

**PONTIFICIA UNIVERSIDAD CATOLICA DEL ECUADOR**

**FACULTAD DE INGENIERIA**

**ESCUELA DE SISTEMAS**

**DISERTACION PREVIA A LA OBTENCION DEL TÍTULO  
DE INGENIERO EN SISTEMAS**

***“IMPLEMENTACION DEL PROCESO DE GESTION DE  
HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE”***

**NOMBRE**

**HECCER BENAVIDES**

**DIRECTOR/A: ING. BEATRIZ CAMPOS**

**QUITO – ECUADOR, 2011**

## **DEDICATORIA**

A mis padres y hermanos, porque creyeron en mí y porque me sacaron adelante, dándome ejemplos dignos de superación y entrega, porque en gran parte gracias a ellos, hoy puedo ver alcanzada mi meta, ya que siempre estuvieron impulsándome en los momentos más difíciles de mi carrera, y porque el orgullo que sienten por mí, fue lo que me hizo ir hasta el final. Va por ellos, por lo que valen, porque admiro su fortaleza y por lo que han hecho de mí.

A mi hermano, quien estuvo conmigo, durante toda mi carrera, apoyándome en todo problema que tuve y siempre a mi lado para salir juntos.

A Iván Villavicencio, actual gerente de la empresa en la que trabaje durante todo el desarrollo del presente trabajo, por todo su apoyo para que logre alcanzar esta meta y por toda su paciencia mientras me encontraba realizando este trabajo.

A mis tutores, por su paciencia y por creer en mí. Sabios consejos supe captar de ellos.

A mi entrenador y compañeros de Rugby – Jibaros PUCE – por todo su apoyo.

Todo este trabajo, no hubiese podido realizarse, sin el apoyo de todos los mencionados, por tal motivo, este trabajo va por ellos.

## **AGRADECIMIENTO**

El presente trabajo es un esfuerzo en el cual, directa o indirectamente, participaron varias personas leyendo, opinando, corrigiendo, teniéndome paciencia, dando ánimo, acompañando en los momentos de crisis y en los momentos de felicidad.

Agradezco a la Ing. Beatriz Campos por haber confiado en mi persona, por la paciencia y por la dirección de este trabajo. Al Ing. Fabián de la Cruz por los consejos, el apoyo y el ánimo que me brindó. Al Ing. Eddy Sánchez por su paciencia y disposición de ayuda en mi tesis.

Gracias también a mis queridos compañeros, que me apoyaron y me permitieron entrar en su vida durante estos casi cinco años de convivir dentro y fuera del salón de clase.

A mis padres y a mi hermano que me acompañaron en esta aventura que significó la ingeniería y que, de forma incondicional, entendieron mis ausencias y mis malos momentos. A mi hermana, que a pesar de la distancia siempre estuvo atenta para saber cómo iba mi proceso. A ti Alexandra, que desde un principio hasta el día hoy sigues dándome ánimo para terminar este proceso.

Gracias a todos los mencionados por haber fomentado en mí el deseo de superación y el anhelo de triunfo en la vida.

Mil palabras no bastarían para agradecerles su apoyo, su comprensión y sus consejos en los momentos difíciles.

A todos, espero no defraudarlos y contar siempre con su valioso apoyo, sincero e incondicional.

# Contenido

<b>RESUMEN EJECUTIVO .....</b>	<b>1</b>
<b>CAPITULO 1: INTRODUCCION .....</b>	<b>2</b>
1.1    Objetivos .....	2
1.1.1    Objetivo General .....	2
1.1.2    Objetivos Específicos .....	2
1.2    Datos de la Institución.....	2
1.3    Funciones del Área de Acción Social Universitaria de la Facultad de Ingeniería de la PUCE ....	6
1.4    Análisis de la Problemática actual en el área de Acción Social de la Facultad de Ingeniería....	8
<b>CAPITULO 2: FUNDAMENTOS TEORICOS.....</b>	<b>12</b>
2.1    Técnicas de desarrollo de Aplicaciones.....	12
2.1.1    Diseño ágil de Aplicaciones (extreme programming) .....	12
2.1.2    RAD (diseño rápido de aplicaciones).....	17
2.1.3    JAD (diseño de aplicaciones por juntas).....	21
2.1.4    Procesos .....	24
2.2    Análisis de herramientas de desarrollo.....	26
2.2.1    Bases de Datos .....	26
2.2.2    Lenguajes de Programación .....	30
2.2.3    Herramientas CASE.....	34
2.3    Arquitectura de Aplicaciones .....	36
2.3.1    Arquitectura Cliente/Servidor .....	36
2.3.2    Arquitectura por 3 Capas .....	38
<b>CAPITULO 3: CASO DE ESTUDIO.....</b>	<b>41</b>
3.1    Introducción .....	41
3.2    Principales Actividades.....	41
3.3    Justificación de las herramientas seleccionadas para la construcción de la aplicación.....	41
3.3.1    Metodología .....	41

3.3.2	Diseño.....	42
3.3.3	Desarrollo .....	42
<b>CAPITULO 4: LEVANTAMIENTO DEL PROCESO.....</b>		<b>45</b>
4.1	Mapa de Procesos .....	45
4.1.1	Cadena de Valor .....	45
4.2	SIPOC.....	45
4.2.1	Proceso.....	45
4.2.2	Diagrama .....	46
<b>CAPITULO 5: CONSTRUCCION DEL PRODUCTO .....</b>		<b>47</b>
5.1	Inicio.....	48
5.1.1	Análisis De Requerimientos.....	48
5.1.2	Especificaciones de Funcionalidad .....	53
5.1.3	Diagramas de Actividades .....	103
5.1.4	Otros Requerimientos .....	120
5.1.5	Estructura del Producto .....	122
5.2	Elaboración.....	130
5.2.1	Prototipo Ejecutable.....	130
5.2.2	Lista de Riesgos .....	132
5.3	Construcción .....	133
5.3.1	Construcción.....	133
5.3.2	Plan de pruebas del sistema.....	133
5.4	Transición .....	139
5.4.1	Seguridades .....	139
5.4.2	Costos .....	139
5.4.3	Despliegue .....	140
5.4.4	Mantenimiento .....	140

**CAPITULO 6: CONCLUSIONES Y RECOMENDACIONES .....141**

6.1 Conclusiones ..... 141

6.1.1 Diseño..... 141

6.1.2 Desarrollo ..... 141

6.1.3 Soporte ..... 141

6.2 Recomendaciones ..... 142

6.2.1 Diseño..... 142

6.2.2 Desarrollo ..... 142

6.2.3 Soporte ..... 142

**BIBLIOGRAFIA.....143**

**GLOSARIO .....146**

## **RESUMEN EJECUTIVO**

La finalidad del presente trabajo, es desarrollar para el Área de Acción Social de la Facultad de Ingeniería de la PUCE, un sistema informático estable, que permita automatizar las tareas desarrolladas diariamente. En la actualidad, es difícil llevar un control manual de los datos manejados por el área, datos como Proyectos a efectuar, estudiantes involucrados, inscripciones en trabajos, etc.

Tomando como base los conocimientos adquiridos durante toda la carrera, se procedió a diseñar los diagramas para el manejo de los procesos, utilizar las herramientas desarrollo necesarias y requeridas por la facultad, y estructurar las pruebas necesarias para desarrollar un sistema que se encargue de manejar todas estas tareas, logrando así, reducir tiempos de respuesta en los procesos mas críticos del área.

## CAPITULO 1: INTRODUCCION

### 1.1 Objetivos

#### *1.1.1 Objetivo General*

*Analizar, diseñar, y desarrollar un sistema para el manejo del proceso de asignación de horas de extensión universitaria para los estudiantes de la Facultad de Ingeniería de la PUCE.*

#### *1.1.2 Objetivos Específicos*

- Analizar la situación actual del proceso a automatizar.
- Diseñar la mejor alternativa de solución para la automatización del proceso.
- Desarrollar una aplicación que automatice el proceso crítico, utilizando software libre.
- Implementar la aplicación (Manual de Usuario y Técnico).
- Capacitar el manejo del software.
- Desarrollar conclusiones y recomendaciones del trabajo.

### 1.2 Datos de la Institución

#### *Nombre*

Área de Acción Social Universitaria de la Facultad de Ingeniería de la Pontificia Universidad Católica del Ecuador.

#### *Actividad*

Brindar información sobre los diferentes proyectos de Acción Social Universitaria y administrar las horas de extensión universitaria de los estudiantes de ingeniería necesarios para poder graduarse.

#### *Ubicación*

Universidad: Referente a la organización de cada facultad de la PUCE.

Facultad de Ingeniería: Primer piso de la Facultad de Ingeniería de la PUCE, Oficina de Acción Social Universitaria.





### *Características*

La Pontificia Universidad Católica del Ecuador, ofrece a sus estudiantes la opción de ayudar a la comunidad mediante proyectos de acción social que son administrados por el Área de Extensión Universitaria. La Facultad de Ingeniería ofrece también esta actividad, llevando siempre un registro de los estudiantes y las actividades que se realizan a favor de comunidades necesitadas.

Los estudiantes pueden participar libremente en cualquier proyecto disponible en del Área de Extensión Universitaria además de proponer nuevos proyectos. De esta manera, se administran las horas que cada estudiante acumula en estas actividades para cumplir un pre-requisito para graduarse.

### *Misión<sup>1</sup>*

#### ➤ *Como Universidad*

Considera misión propia el contribuir, de un modo riguroso y crítico, a la tutela y desarrollo de la dignidad humana y de la herencia cultural, mediante la investigación, la docencia y los diversos servicios ofrecidos a las comunidades locales, nacionales e internacionales.

En dicha misión, asume el deber de prestar particular atención a las dimensiones éticas de todos los campos del saber y del actuar humano, tanto a nivel individual como social.

En este marco propugna el respeto a la dignidad y a los derechos de la persona humana, y a sus valores trascendentes, y apoya y promueve la implantación de la justicia en todos los órdenes de la existencia.

Goza de aquella autonomía institucional que le es necesaria para cumplir sus funciones eficazmente.

---

<sup>1</sup> Información tomada del Plan Estratégico de Desarrollo Institucional 2008-2013, de la PUCE

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

Garantiza a sus miembros la libertad académica, salvaguardando los derechos de la persona y de la comunidad dentro de las exigencias de la verdad y del bien común.

Dirige su actividad hacia la persona integral, para superar una formación meramente profesional. Por ello trata de formar a sus miembros intelectual y moralmente, para el servicio a la sociedad.

Examina a fondo la realidad con los métodos propios de cada disciplina académica, estableciendo después un diálogo entre las diversas disciplinas que las enriquezca mutuamente. Con ello pretende la integración del saber.

Promueve el compromiso de todos los miembros de la comunidad universitaria para la consecución de los fines institucionales, a través del diálogo y la participación.

➤ *Como Universidad Católica*

Se inspira en los principios cristianos; propugna la responsabilidad del ser humano ante Dios, el respeto a la dignidad y derechos de la persona humana y a sus valores trascendentales; apoya y promueve la implantación de la justicia en todos los órdenes de la existencia; propicia el diálogo de las diversas disciplinas con la fe, la reflexión sobre los grandes desafíos morales y religiosos, y la praxis cristiana.

➤ *Como Universidad dirigida por la Compañía de Jesús*

Promueve la implantación y el desarrollo de la pedagogía ignaciana en todas sus actividades académicas.

***Visión***

En los próximos años, la PUCE, fundamentada en el pensamiento y en las directrices pedagógicas ignacianas, se consolidará como un sistema nacional integrado competitivo y autosostenible, con infraestructura tecnológica de vanguardia.



## IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

Será reconocida por su gestión ética en servicio de la comunidad, y por su estructura académica moderna para la formación de profesionales con responsabilidad social.

Será también reconocida por los resultados de la investigación científica desarrollada en sus unidades académicas, por realizar su gestión con el apoyo de un sistema técnico, innovador y efectivo, con procesos eficientes y recursos humanos capacitados y comprometidos con la misión institucional.

### *Valores Institucionales de la PUCE<sup>2</sup>*

- Justicia
- Integridad
- Responsabilidad social
- Equidad
- Innovación
- Igualdad de oportunidades
- Diversidad
- Reconocimiento del mérito individual
- Sentido de pertenencia a la Institución
- Orientación de servicio
- Mejoramiento continuo
- Trabajo en equipo
- Puntualidad
- Disciplina

### *Objetivos Estratégicos de la PUCE*

1. Lograr que la PUCE disponga de una estructura académica moderna acorde con la demanda de la sociedad.

---

<sup>2</sup> Información tomada del Plan Estratégico de Desarrollo Institucional 2008-2013, de la PUCE



2. Fortalecer el sistema de investigación que contribuya a la innovación y al desarrollo del pensamiento, la ciencia y la tecnología.
3. Fortalecer el sistema de interacción con la comunidad basado en los principios y valores de la PUCE.
4. Implantar un Sistema de Gestión administrativa y Financiera que contribuya de manera efectiva al desarrollo académico.
5. Lograr la sustentabilidad económica para el funcionamiento de la PUCE.
6. Consolidar y fortalecer el SINAPUCE.
7. Mejorar el entorno laboral y las capacidades del personal académico y administrativo.

### **1.3 Funciones del Área de Acción Social Universitaria de la Facultad de Ingeniería de la PUCE**

El Área de Acción Social Universitaria es responsable principalmente de llevar el control de las horas de Extensión Universitaria de cada estudiante, además de esto, se encarga de las siguientes actividades:

- Informar a los estudiantes de los diferentes proyectos que pueden realizar.
- Llevar a cabo seminarios respecto a temas comunitarios.
- Establecer vínculos entre las organizaciones externas y la Universidad, para la colaboración de sus estudiantes.
- Emitir fichas de finalización de horas de extensión universitaria. Pre-requisito para la graduación del estudiante.
- Informar a los estudiantes el estado actual de sus horas de Extensión Universitaria.

#### ***Normas de Funcionamiento del Área de Extensión Social Universitaria***

Para cumplir con los objetivos del Área de Extensión Social Universitaria, se establecen las siguientes normas para su funcionamiento:

1. Todo estudiante de la Facultad de Ingeniería de la Pontificia Universidad Católica del Ecuador tiene la obligación de cumplir en labores de “ACCION

SOCIAL UNIVERSITARIA DE PROMOCION COMUNITARIA” llamada también EXTENSION SOCIAL UNIVERSITARIA, un tiempo total de 200 horas y justificar su cumplimiento previamente a la Disertación de Grado.

2. Los trabajos de Acción Social Universitaria deben realizar los estudiantes que han cumplido el primer nivel de estudios, para lo cual deben inscribirse obligatoriamente en las oficinas de Extensión Social Universitaria de la Facultad de Ingeniería, con el Coordinador Ing. Eddy Sánchez S.

Las acciones que se pueden realizar y que deben ser previamente calificadas por el Coordinador de Extensión Universitaria son las siguientes:

*Labores de campo en proyectos de Acción Social de Promoción Comunitaria tales como:*

- Investigación y conocimiento de realidades sociales en comunidades urbanas o rurales.
- Capacitación y talleres según sus necesidades prioritarias.
- Coparticipación con estudiantes y otras unidades académicas de la PUCE.
- Talleres de conocimiento general.

*Ingeniería Civil:*

- Asistencia técnica en:
  1. Proyectos referentes a estructuras, obras básicas de topografía, vialidad, agua potable, mecánica de suelos, alcantarillado, construcciones, investigación de materiales, etc.
  2. Ordenamiento y tipificación de asentamientos humanos en relación directa con su región: viviendas dispersas o concentradas, servicios comunales, centros de servicios o de desarrollo.
  3. Análisis de tecnologías apropiadas y sus aplicaciones.

*Ingeniería de Sistemas:*

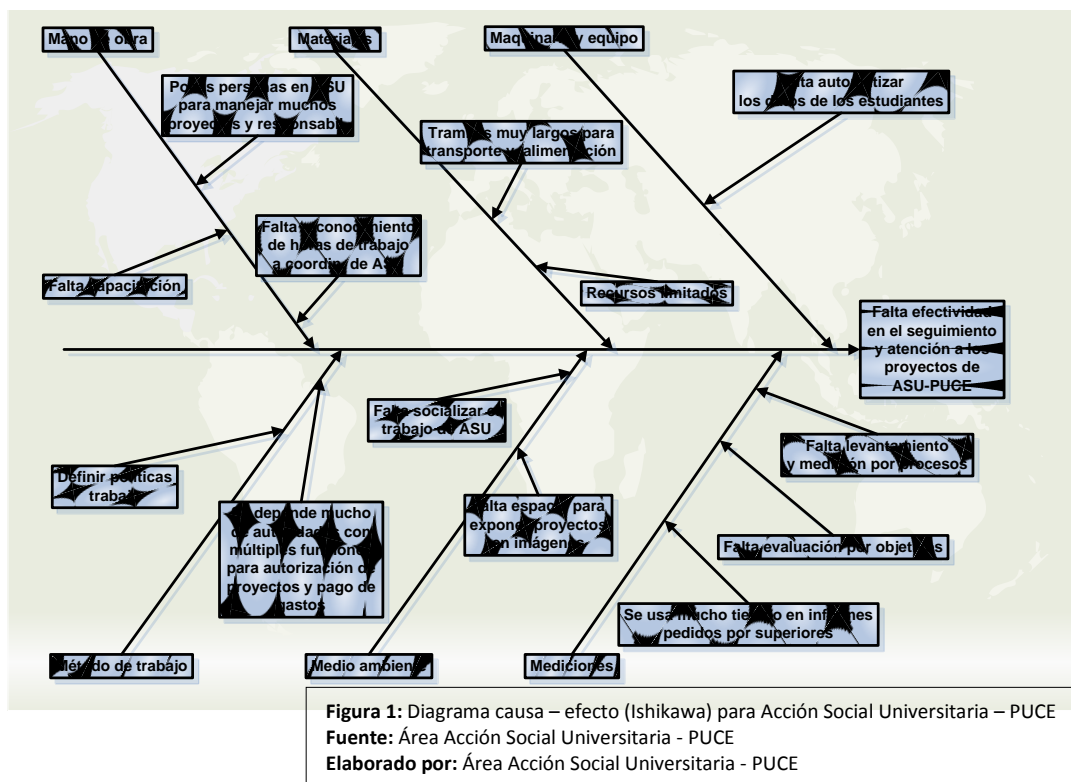
- Asistencia técnica en:
  1. Proyectos vinculados a satisfacer las necesidades de la informática de las comunidades urbanas y rurales del Ecuador, cuyo objetivo sea el bienestar social.

2. Los proyectos deben relacionarse con el diseño, la implementación y automatización de sistemas informáticos y de comunicación electrónica.
  3. Capacitación y apoyo técnico a personas de las comunidades en las herramientas de la informática tales como: manejo y administración de sistemas, utilización de bases de datos para organizar y planificar actividades de la comunidad, procesadores de texto, hojas electrónicas, etc.
  4. Análisis de tecnologías informáticas apropiadas y sus aplicaciones en las comunidades.
- 
- Los estudiantes a quienes se les asigne un proyecto de acción social universitaria, presentarán al inicio una síntesis del alcance del mismo y a la finalización, un informe personal técnico, en el que se incluyan la valoración social de la coparticipación, los planos, estudios y más documentos justificativos.
  - Los señores profesores que requieran la colaboración de estudiantes vía extensión social universitaria, escogerán al candidato y solicitarán por escrito la participación del mismo.
  - El profesor tiene la obligación de llevar el control en el respectivo formulario de cumplimiento de las tareas asignadas al estudiante.

#### **1.4 Análisis de la Problemática actual en el área de Acción Social de la Facultad de Ingeniería**

Se usó esta herramienta (diagrama de Ishikawa) con el fin de determinar causas que disminuyen la eficiencia y rendimiento en la atención y seguimiento a los proyectos de ASU – PUCE y transformarlos en objetivos a ser superados en el año 2010, tal como se indica en la Figura 1.

## IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE



### Objetivo Principal:

Mejorar el desempeño, seguimiento y atención a los proyectos de Acción Social Universitaria ASU – PUCE con coparticipación de la comunidad universitaria y la comunidad.

### Objetivos Secundarios:

- Automatizar los datos de ASU – PUCE
- Presupuestar los egresos periódicos de los proyectos de ASU – PUCE y comparar con la partida que tiene la Universidad para proyectos de acción social y priorizar gastos.
- Reconocer horas de su carga de trabajo, a todos los profesores coordinadores de ASU – PUCE en cada Facultad, de acuerdo a un informe por proyecto.
- Evaluar la carga de trabajo del personal administrativo de ASU – PUCE (Ec. Adriana Jaramillo) y de ser el caso permitir el trabajo de becarios – ayudantes, conforme a los reglamentos de la PUCE.
- Promover la capacitación de los miembros de ASU –PUCE para mejorar su desempeño.

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

- Definir políticas de trabajo y un reglamento que incluya horarios de trabajo de campo que permita al personal de ASU – PUCE contactarse con la comunidad y hacer un seguimiento de sus proyectos.
- Socializar el trabajo de ASU – PUCE, tanto en el medio interno como externo.
- Levantar los procesos de ASU – PUCE y trabajar en base a objetivos por desempeño.

***Recomendaciones.***

- La dirección de ASU – PUCE debe ser ejercida en base a comprobada experiencia en proyectos de Acción y Extensión Social Universitaria.
- Disminuir los pasos que se deben seguir para autorización de pagos de transporte, alimentación etc. de salidas de ASU - PUCE.

***Estrategias propuestas.***

- Independencia de ASU – PUCE de la Dirección General de Estudiantes (DGE)
- Dependencia organizativa del Rectorado y Vicerrectorado similar a direcciones como Dirección de Pastoral Universitaria y Dirección de Relaciones Internacionales, entre otras.
- Coordinar acciones y proyectos de ASU – PUCE con la Comisión de Vinculación con la Colectividad conforme al capítulo V. Artículos 37,38 y 39 del Estatuto de la PUCE.

***Problemas respecto al tratamiento de la Información***

El tratamiento de la información de cada estudiante con respecto a sus trabajos de Acción Social Universitaria, así como los detalle de cada proyecto o servicio, no son fácilmente accesibles, ya que son archivados manualmente y toma tiempo manejarlos.

Además, la forma en que se almacena la información no es del todo automatizada, debida a que solamente utilizan herramientas de ofimática que a la larga no son muy eficientes.





## **IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE**

Es decir, no existe un software que permita gestionar la información de este proceso de manera segura, ágil, y oportuna.

Actualmente, el Ingeniero Eddy Sánchez es el Director del Área de Acción Social Universitaria de la Facultad de Ingeniería de la PUCE, quien ha realizado el requerimiento para automatizar su proceso más crítico que es la gestión de horas de Extensión Universitaria para los estudiantes de la Facultad.

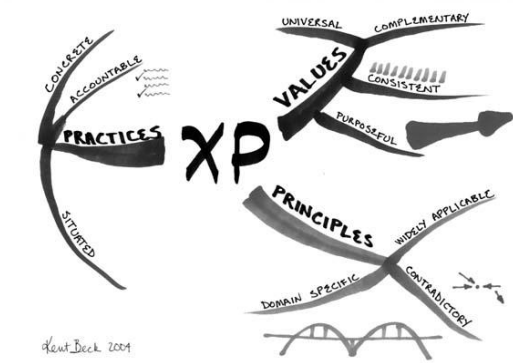
El área solo utiliza hojas electrónicas para llevar el registro de cada estudiante con sus proyectos, y como respaldo tienen el archivo (pilas y pilas de carpetas con fichas) que muestra la información detallada de cada estudiante. Esta manera de llevar la información no es la más óptima en la actualidad.



## CAPITULO 2: FUNDAMENTOS TEORICOS

### 2.1 Técnicas de desarrollo de Aplicaciones

#### 2.1.1 Diseño ágil de Aplicaciones (*extreme programming*)



La programación extrema o eXtreme Programming (XP) es un enfoque de la ingeniería de software formulado por Kent Beck, autor del primer libro sobre la materia, *Extreme Programming Explained: Embrace Change* (1999).

Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos.

Se puede considerar la programación extrema como la adopción de las mejores metodologías de desarrollo de acuerdo a lo que se pretende llevar a cabo con el proyecto, y aplicarlo de manera dinámica durante el ciclo de vida del software.



**Figura 2:**  
Ciclo de vida diario XP  
**Fuente:**  
<http://www.extremeprogramming.org>  
**Elaborado por:**  
<http://www.extremeprogramming.org>

#### Objetivos

Los objetivos de XP son muy simples: la satisfacción del cliente. Esta metodología trata de dar al cliente el software que él necesita y cuando lo necesita. Por tanto, se debe responder muy rápido a las necesidades del cliente, incluso cuando los cambios sean al final de ciclo de la programación.

El segundo objetivo es potenciar al máximo el trabajo en grupo. Tanto los jefes de proyecto, los clientes y desarrolladores, son parte del equipo y están involucrados en el desarrollo del software.

### *Las 4 actividades básicas*

#### *Codificar*

Es la única actividad de la que no se puede prescindir. Sin código fuente no hay programa, aunque hay gente que cuenta que existe software en producción del que se perdió el código fuente. Por tanto se necesita codificar y plasmar las ideas a través del código.



#### *Hacer pruebas*

Las características del software que no pueden ser demostradas mediante pruebas simplemente no existen. Las pruebas dan la oportunidad de saber si lo que se implementó es lo que en realidad se pensaba que debía ser implementado. Las pruebas indican que el trabajo realizado funciona, cuando no se puede pensar en ninguna prueba que pudiese originar un fallo en nuestro sistema entonces se ha acabado por completo.

#### *Escuchar*

Los programadores no lo conocen todo, y sobre todo muchas cosas que las personas de negocios piensan que son interesantes. Si ellos pudieran programarse su propio software ¿para qué nos necesitarían?

Si se van a hacer pruebas, hay que preguntar si lo obtenido es lo deseado, y se tiene que preguntar quién necesita la información. Se debe escuchar de los clientes, cuales son los problemas de su negocio, se debe tener una habilidad de escuchar muy activa, tratando de explicar lo que es fácil y difícil de obtener.



#### *Diseñar*

El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben de ser sencillos, si alguna parte del sistema es de desarrollo complejo, hay que dividirla en varias. Si hay fallos en el diseño o malos diseños, éstos deben ser corregidos cuanto antes.

### *Fases de la metodología XP*

#### **Planificación.**

XP plantea la planificación como un permanente diálogo entre las partes la empresarial (deseable) y la técnica (posible). Las personas del negocio necesitan determinar:

- **Ámbito:** ¿Qué es lo que el software debe de resolver para que este genere valor?
- **Prioridad:** ¿Qué debe ser hecho en primer lugar?
- **Composición de versiones:** ¿Cuánto es necesario hacer para saber si el negocio va mejor con software que sin él? En cuanto el software aporte algo al negocio se debe tener listas las primeras versiones.
- **Estimaciones:** ¿Cuánto tiempo lleva implementar una característica?
- **Consecuencias:** Informar sobre las consecuencias de la toma de decisiones por parte del negocio. Por ejemplo el cambiar las bases de datos a Oracle.
- **Procesos:** ¿Cómo se organiza el trabajo y el equipo?



#### **Pequeñas versiones**

Cada versión debe ser tan pequeña como fuera posible, conteniendo los requisitos de negocios más importantes, las versiones deben tener sentido como un todo, es decir, no se puede implementar media característica y lanzar la versión.

#### **Diseño**

##### **Metáfora.**

Una metáfora es una historia que todo el mundo puede contar acerca de cómo funciona el sistema. Algunas veces se podrá encontrar metáforas sencillas “Programa de gestión de compras, ventas, con gestión de cartera y almacén”. Las metáforas ayudan a cualquier persona a entender el objeto del programa.



##### **Diseño sencillo.**

El diseño adecuado para el software es aquel que:

- Funciona con todas las pruebas.

- No tiene lógica duplicada.
- Manifiesta cada intención importante para los programadores
- Tiene el menor número de clases y métodos.

### ***Desarrollo.***

#### **Recodificación.**

Cuando se implementa nuevas características en los programas, se plantea la manera de hacerlo lo más simple posible, después de implementar esta característica, hay que preguntarse cómo hacer el programa más simple sin perder funcionalidad, este proceso se le denomina recodificar o refactorizar (refactoring). Esto a veces nos puede llevar a hacer más trabajo del necesario, pero a la vez se estará preparando el sistema para que en un futuro acepte nuevos cambios y pueda albergar nuevas características.



#### **Programación por parejas.**

Todo el código de producción lo escriben dos personas frente al ordenador, con un sólo ratón y un sólo teclado. Cada miembro de la pareja juega su papel: uno codifica en el ordenador y piensa la mejor manera de hacerlo, el otro piensa más estratégicamente, ¿Va a funcionar?, ¿Puede haber pruebas donde no funcione?, ¿Hay forma de simplificar el sistema global para que el problema desaparezca?

#### **Propiedad colectiva.**

Cualquiera que crea que puede aportar valor al código en cualquier parcela puede hacerlo, ningún miembro del equipo es propietario del código. XP propone una propiedad colectiva sobre el código, nadie conoce cada parte igual de bien, pero todos conocen algo sobre cada parte, esto ayudará a la preparación para la sustitución no traumática de cada miembro del equipo.

#### **Integración continúa.**

El código se debe integrar como mínimo una vez al día, y realizar las pruebas sobre la totalidad del sistema. Una pareja de programadores se encargará de integrar todo el código en una máquina y de realizar todas las pruebas hasta que estas funcionen totalmente.

### **Cliente In-situ.**

Un cliente real debe sentarse con el equipo de programadores, estar disponible para responder a sus preguntas, resolver discusiones y fijar las prioridades. Lo difícil es que el cliente ceda una persona que conozca el negocio para que se integre en el equipo. Normalmente estos elementos son muy valiosos, pero hay hacerles ver que será mejor para su negocio tener un software pronto en funcionamiento, y esto no implica que el cliente no pueda realizar cualquier otro trabajo.

### **Pruebas**

#### **Hacer pruebas.**

No debe existir ninguna característica en el programa que no haya sido probada, los programadores escriben pruebas para chequear el correcto funcionamiento del programa, los clientes realizan pruebas funcionales. El resultado: un programa más seguro que conforme pasa el tiempo, es capaz de aceptar nuevos cambios.



### **Observaciones de la Metodología**

Se critica el mito de las 40 horas semanales, y que es un lujo para las exigencias del mercado.

XP está diseñado para grupos de pequeños programadores, más de 10 ya sería muy complicado, y peor, para que estén en el mismo centro de trabajo.

XP supone:

- Las personas son claves en los procesos de desarrollo.
- Los programadores son profesionales no necesitan supervisión.
- Los procesos se aceptan y se acuerdan, no se imponen.
- Desarrolladores y gerentes comparten el liderazgo del proyecto.
- El trabajo de los desarrolladores con las personas que conocen el negocio es regular, no puntual.

Y conviene recordar que:

*“Ninguna metodología hace el trabajo por sí sola, pero te podrá ayudar”.*



### 2.1.2 RAD (diseño rápido de aplicaciones)

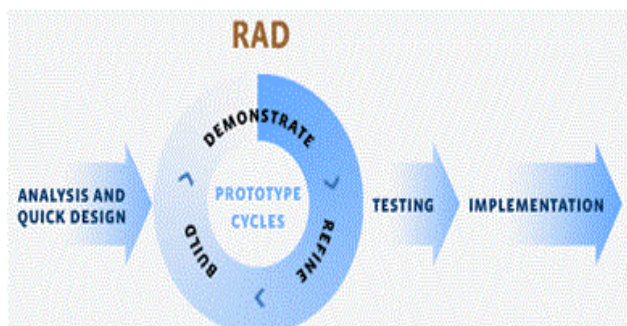


Figura 3: Ciclo de vida RAD

Fuente: <http://www.etondigital.com>

Elaborado por: <http://www.etondigital.com>

El Desarrollo Rápido de Aplicaciones (Rapid Application Development RAD) es un modelo de proceso del desarrollo del software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto. RAD es una adaptación a "alta velocidad"

en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes.

Si se comprenden bien los requisitos y se limita el ámbito del proyecto, el proceso RAD permite al equipo de desarrollo crear un sistema completamente funcional dentro de periodos cortos de tiempo. Cuando se utiliza principalmente para aplicaciones de sistemas de información, el enfoque RAD comprende las siguientes fases:

#### Modelado de gestión

El flujo de información entre las funciones de gestión se modela de forma que responda a las siguientes preguntas: ¿Qué información conduce el proceso de gestión? ¿Qué información se genera? ¿Quién la genera? ¿A dónde va la información? ¿Quién la procesó?

#### Modelado de datos

El flujo de información definido como parte de la fase de modelado de gestión se refina como un conjunto de objetos de datos necesarios para apoyar la empresa. Se definen las características (llamadas atributos) de cada uno de los objetos y las relaciones entre estos objetos.

#### Modelado de proceso

Los objetos de datos definidos en la fase de modelado de datos quedan transformados para lograr el flujo de información necesario para implementar una función de gestión. Las descripciones del proceso se crean para añadir, modificar, suprimir, o recuperar un objeto de datos.



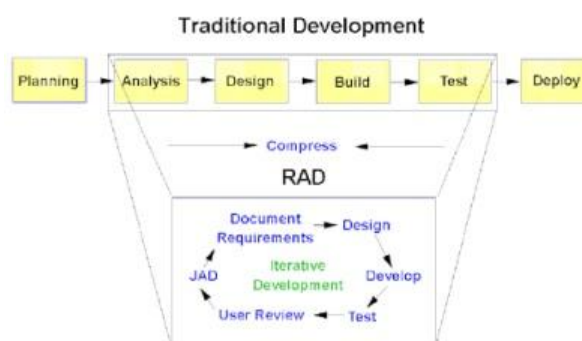
## Generación de aplicaciones

RAD asume la utilización de técnicas de cuarta generación. En lugar de crear software con lenguajes de programación de tercera generación, el proceso RAD trabaja para volver a utilizar componentes de programas ya existentes (cuando es posible) o a crear componentes reutilizables (cuando sea necesario). En todos los casos se utilizan herramientas automáticas para facilitar la construcción del software.

## Pruebas de entrega

Como el proceso RAD enfatiza la reutilización, ya se han comprobado muchos de los componentes de los programas. Esto reduce tiempo de pruebas. Sin embargo, se deben probar todos los componentes nuevos y se deben ejercitar todas las interfaces a fondo.

No todos los tipos de aplicaciones son apropiados para RAD. Si un sistema no se puede modular adecuadamente, la construcción de los componentes necesarios para RAD será problemático. Si está en juego el alto rendimiento, y se va a conseguir el rendimiento convirtiendo interfaces en componentes de sistema, el enfoque RAD puede que no funcione.



**Figura 4:** Desarrollo tradicional vs RAD (ciclo de vida)  
**Fuente:** <http://www.casemaker.com>  
**Elaborado por:** <http://www.casemaker.com>

## Aspectos esenciales de RAD



El desarrollo rápido de aplicaciones tiene 4 aspectos esenciales: **metodología, gente, administración, y herramientas.** Si alguno de estos ingredientes es inadecuado, el desarrollado no será



de alta velocidad. Los ciclos de vida del desarrollo, que rodean estos ingredientes juntos lo más efectivamente posible, son los de mayor importancia.

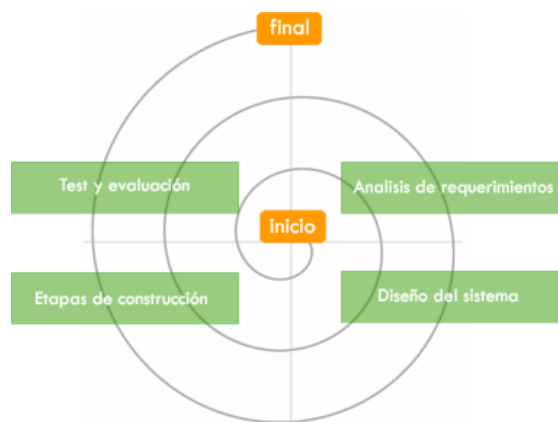
## Metodología

RAD ataca proporciona un medio para el desarrollo de sistemas más rápido, mientras se reducen los costos y se aumenta la calidad. Aspectos fundamentales de la metodología RAD, incluye:

- La combinación de las mejores técnicas y especificación de la secuencia de las tareas que harán estas técnicas más efectivas.
- El uso de prototipos de evolución constante que eventualmente son transformados en el producto final
- El uso de juntas (talleres) en vez de entrevistas, para recolectar requerimientos y revisión de diseños.
- La selección de un grupo de herramientas CASE para soportar el modelado, prototipos, y la re-usabilidad de código, como también la automatización de muchas de las combinaciones de técnicas.



El ciclo de vida, que se ilustra en la figura 5, incluye todas las actividades y tareas necesarias para definir y alcanzar las necesidades del negocio, y diseñar, desarrollar e implementar el sistema de aplicación que es compatible con estos requisitos.



**Figura 5:** Etapas del ciclo de vida RAD  
**Fuente:** <http://www.casemaker.com>  
**Elaborado por:** <http://www.casemaker.com>

### ***Planificación de Requerimientos***

También se conoce como la etapa de Definición de Conceptos, esta etapa define las funciones del negocio y los datos de las materias que el sistema soportara, determina también el alcance del sistema.

### ***Diseño del Usuario***

También conocida como la etapa de Diseño Funcional, esta etapa utiliza talleres para modelar los datos y procesos del sistema, y construir un prototipo de trabajo de los componentes críticos del sistema.

### ***Construcción***

También se conoce como la fase de Desarrollo, esta etapa completa la construcción del sistema de aplicación física, construye el sistema de conversión, y desarrolla ayudas usuario y planes de ejecución de trabajo.

### ***Implementación***

También conocida como la etapa de Deployamiento (nombre inglés para implementación), esta etapa incluye pruebas de usuario final y capacitación, conversión de datos, y la construcción del sistema de aplicación.



#### **Gente**

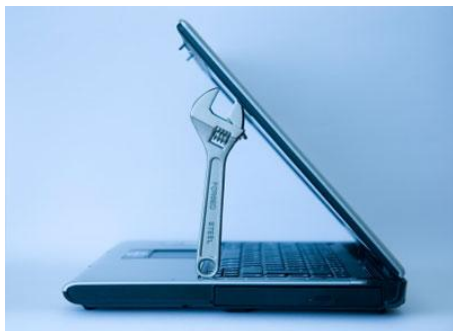
El éxito del desarrollo rápido de aplicaciones depende de la participación de las personas con las habilidades y talentos correctos. Unas herramientas excelentes son esenciales para un rápido desarrollo de aplicaciones, sin embargo estas, por sí mismas, no garantizan el éxito.

### **Administración**

Lograr un desarrollo de alta velocidad es un proceso complejo. Los sistemas no serán desarrollados e implementados rápidamente si la burocracia y los obstáculos políticos se interponen, o si los usuarios no están debidamente involucrados. La gestión debe ser totalmente comprometida a RAD con el fin de gestionar el cambio en la cultura. Una buena gestión y la dedicación a los ideales del



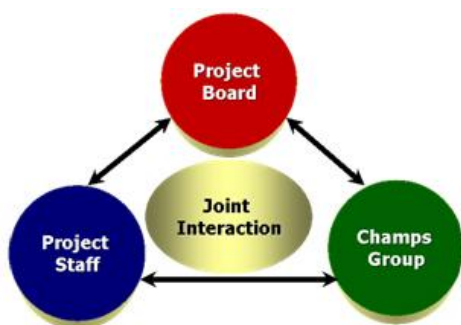
desarrollo rápido de aplicaciones son esenciales para la construcción más rápida del sistema.



### Herramientas

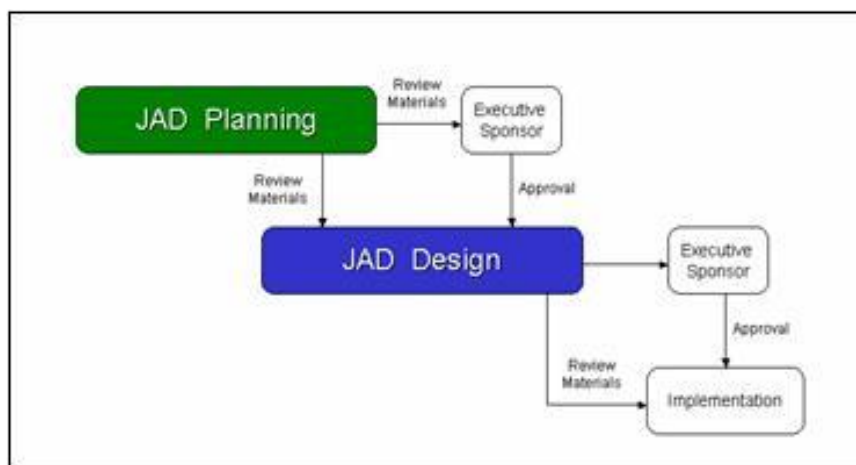
La metodología RAD utiliza tanto las herramientas informáticas como las técnicas de los humanos para alcanzar los objetivos de alta velocidad y alta calidad. Lo anterior ha sido principalmente relacionado con los objetivos del RAD y el papel de las organizaciones y las personas dentro de esas organizaciones en el logro de las metas. El éxito de cualquier proyecto RAD es principalmente dependiente en las herramientas utilizadas. Estas herramientas pueden cambiar los métodos de construcción mediante el uso de técnicas de automatización de diseño, generación de código, y planificación y análisis asistido por ordenadores. Las herramientas de software utilizadas en RAD son Computer-Aided Systems Engineering (CASE). Véase sección 2.2.3.

#### 2.1.3 JAD (diseño de aplicaciones por juntas)



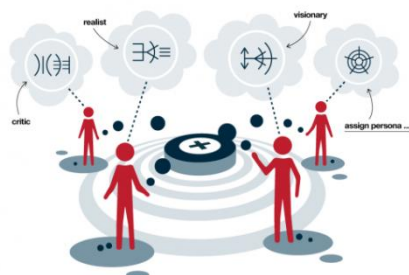
JAD (Joint Application Development) es una técnica de definición de requisitos y de diseño de la interfaz de usuario, basada en reuniones participativas entre clientes, directiva y desarrolladores. En dichas reuniones los temas a tratar se centran más en el negocio que en el asunto técnico. Lógicamente está más orientado a proyectos de cliente (o bien sistemas a medida, como también se los conoce), y permite recolectar requisitos eficientemente.

Hay que tener cuidado porque estas reuniones pueden hacer ver a los clientes una falsa realidad en cuanto al progreso del proyecto o la productividad. Además, hay que prestar especial cuidado con las estimaciones tempranas, aquellas que entrañan un mayor riesgo por el mayor desconocimiento del sistema y que deben ofrecer una amplitud de rango mayor entre mejor estimación y estimación pesimista.



**Figura 6:** Fases del ciclo de vida JAD  
**Fuente:** <http://es.debugmodeon.com>  
**Elaborado por:** <http://es.debugmodeon.com>

Como se ilustra en la figura 6, JAD consta de dos fases principales: planificación y diseño. Ambas tratan los requisitos, pero a distinto nivel de abstracción. Si bien en planificación se tratan los requisitos a un nivel más alto, estudiando sobre todo la utilidad y la viabilidad de los mismos, en la fase de diseño se realiza un uso intensivo de prototipos y se diseña la interfaz de usuario, el presupuesto, la calendarización y el esquema de la base de datos.



Cada una de estas fases llevaría en torno a entre uno y diez días. No confundir la fase diseño-JAD con la fase de diseño del proyecto; JAD es una técnica que se aplicaría en fase de planificación y análisis.

Una vez terminado el proceso de JAD, *se sigue con el modelo de desarrollo elegido*. Es decir, JAD es independiente del modelo de desarrollo, con lo cual es aplicable siempre.

### ***Fundamentos del J.A.D***

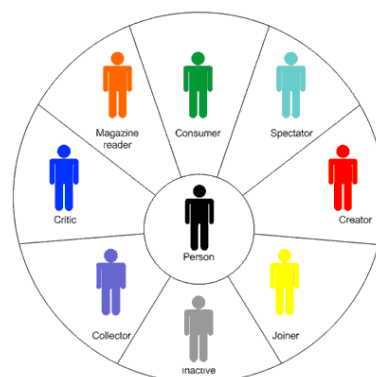
El proceso de JAD se basa en cuatro ideas simples:

1. La gente que hace un trabajo tiene la mejor comprensión de ese trabajo.
2. La gente entrenada en Tecnologías de la Información tiene la mejor comprensión de las posibilidades de esas tecnologías.
3. Los sistemas de información y los procesos del negocio raramente existen en forma aislada, más bien trascienden los límites de cualquier sistema u oficina y afectan el trabajo en departamentos relacionados. La gente que trabaja en estas áreas

relacionadas tiene una percepción valiosa del papel del sistema dentro de una comunidad más amplia.

4. Los mejores sistemas de información se diseñan cuando todos estos grupos trabajan juntos en un proyecto como socios iguales.

Puede trazarse un paralelo entre lo que el proceso de JAD le da al desarrollo de sistemas informáticos y lo que Henry Ford le dio a la fabricación de los automóviles: un método para organizar la maquinaria, los materiales, y el trabajo de modo de poder ensamblar un automóvil en forma más barata y rápida que nunca antes (la línea de montaje). La meta en el desarrollo de sistemas es identificar lo que necesitan los usuarios y después fijar un sistema o proceso que lo proporcionen. Los métodos tradicionales tienen varios factores de retraso incorporados que empeoran cuanto más gente está implicada en el proyecto.



### ***Roles del JAD***

#### ***El patrocinador del proyecto***



Es quien presupuesta el proyecto, el dueño del sistema. Tienen el lugar más alto en la organización, de modo que ellos pueden tomar las decisiones y proporcionar los recursos necesarios y apoyar para el proyecto.

#### ***Líder del Proyecto***

Tiene que estar comprometido al proyecto, tener un conocimiento de fondo del área comercial y sistemas de información actuales relacionados.



#### ***Registrador***



Toma los apuntes durante una sesión, y entonces los revisa en un resumen conciso de discusiones y decisiones. Es importante que las notas resultantes no son una transcripción de quién lo dijo. Este papel puede compartirse entre varios miembros del equipo según necesidad. Estas

notas sirven como una referencia al grupo al retomar las discusiones, y para la referencia del retorno en los puntos complejos.

### ***Time keeper***

Son los responsables de asegurar que se cumpla la agenda establecida a fin de optimizar el tiempo



### ***Cientes***



Son los que conocen cómo funcionara el sistema y cómo se usa. Ellos ayudarán al equipo a comprender las tareas manipuladas por el sistema.

### ***Observaciones de la Metodología***

Comparado con métodos tradicionales, JAD es más costoso y puede ser incómodo si el grupo es demasiado grande concerniente al tamaño del proyecto. Muchas compañías encuentran, sin embargo, que JAD permite que los usuarios dominantes participen con eficacia en los requisitos que modelan proceso.

Cuando los usuarios participan en el proceso del desarrollo de los sistemas, es más probable percibir un sentido de la propiedad en los resultados, y la ayuda para el nuevo sistema. Cuando se está utilizado correctamente JAD, se puede dar lugar a una declaración más exacta de los requisitos del sistema, y a una comprensión mejor de metas comunes.

#### ***2.1.4 Procesos***

##### ***Mapa de Procesos***



Un mapa de procesos permite tener una visión global del sistema, en este caso de lo que se denomina la organización (la empresa). Visualiza la relación entre la organización y las partes interesadas.

Permite obtener una primera idea sobre las operaciones, las funciones y los procesos. Debe también, representar las relaciones e interrelaciones dentro de la organización y con las partes interesadas.



Para realizar el mapa de procesos de la organización, lo primero que debe hacerse es:

- *Identificar quienes son los dueños, los clientes y los proveedores.*
- *Plantear cual es el objetivo a alcanzar.*
- *Qué y quien da impulso al proceso.*
- *Cuáles son los elementos de entrada del proceso.*
- *Como y a través de quien (responsable) y con quien (interrelaciones) se ejecuta el proceso.*
- *Cuáles son los resultados del proceso (salidas).*
- *Cómo y cuando se mide, visualiza y evalúa la aptitud de funcionamiento.*
- *Visualizar que el proceso es claro y comprensible (realización de un flujo grama).*
- *Evidenciar que el cliente está satisfecho.*

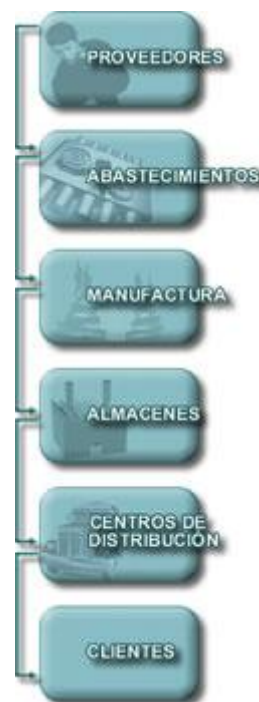
Hay que clasificar los procesos, preparar un modelo de proceso para la empresa y preparar la documentación de los procesos (descripción y flujograma).

### ***Cadena de Valor***

En un entorno donde cada vez es más difícil competir, la eficiencia en la cadena de valor nos funciona como un impulsor para disminuir costos y usar eficientemente los recursos, sin embargo, su objetivo principal es establecer esquemas logísticos, operativos y estratégicos que rompan con los tradicionales, para así establecer fuertes ventajas competitivas en el mediano y largo plazo.

El concepto de cadena de valor se enfoca en la identificación de los procesos y operaciones que aportan valor al negocio, desde la creación de la demanda hasta que ésta es entregada como producto final.

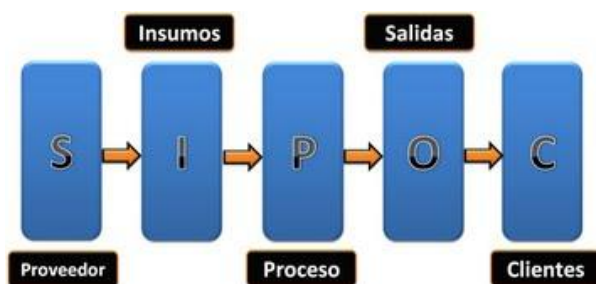
Se encuentra formada por dos subsistemas: la cadena de demanda, que se refiere a todo los procesos relacionados con la creación y entendimiento de la demanda; y la cadena de suministros, que se refiere a alinear todos los procesos del negocio hacia el surtimiento de los requerimientos de la demanda en tiempo, cantidad y forma; es decir, lograr la excelencia en la ejecución logística obteniendo altos niveles de servicio al costo más bajo.



### S.I.P.O.C

SIPOC es una foto de alta definición del proceso que describe cómo mismo es brindado al cliente. Es un acrónimo de Suppliers (Proveedores) - Inputs (Entradas) - Process (Proceso) - Outputs (Productos) - Customers (Clientes). La definición de cada una de estas entidades SIPOC figura a continuación.

- Los **Proveedores** ofrecen insumos para el proceso.
- Los **Insumos** definen el material, servicio y/o la información que utiliza el proceso para producir los resultados.
- El **proceso** es una secuencia definida de actividades, por lo general agrega valor a los insumos para producir los resultados para los clientes.
- Los **resultados** son los productos, servicios, y / o información que tiene valor para los clientes.
- Los **clientes** son los usuarios de los productos producidos por el proceso.



En términos más formales, SIPOC puede ser visto como un mapa de alta definición del proceso. Generalmente se usa durante la fase de definición de un proyecto de mejora de procesos, ya que nos ayuda a entender claramente el propósito y el

alcance de un proceso. Nos da una visión inicial para los insumos vitales (o variables X), de un proceso [ $Y = f(X)$ ] que tienen un impacto significativo en los productos críticos resultados (o variables Y).

## 2.2 Análisis de herramientas de desarrollo

### 2.2.1 Bases de Datos



Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular.

El término de bases de datos fue escuchado por primera vez en 1963, en un simposio celebrado en California, USA.



Una base de datos se puede definir como un conjunto de información relacionada que se encuentra agrupada ó estructurada.

Desde el punto de vista informático, la base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos.

### ***Características***

Entre las principales características de los sistemas de base de datos podemos mencionar:

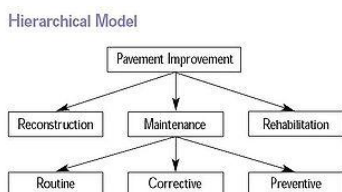
- Independencia lógica y física de los datos.
- Redundancia mínima.
- Acceso concurrente por parte de múltiples usuarios.
- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar.

En una base de datos, las entidades y atributos del mundo real, se convierten en registros y campos. Estas entidades pueden ser tanto objetos materiales como libros o fotografías, pero también personas, incluso conceptos e ideas abstractas. Las entidades poseen atributos y mantienen relaciones entre ellas.

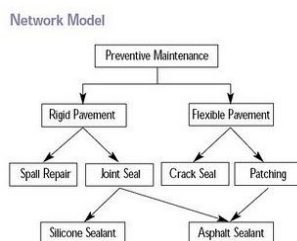
### ***Modelos de tratamiento de datos***

Los modelos clásicos de tratamiento de los datos son los que se presentan en la figura 7.

#### ***Jerárquico***

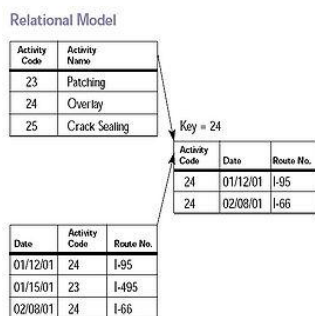


#### ***En red***

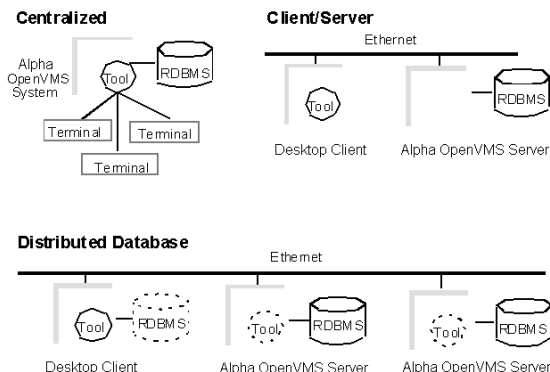


# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

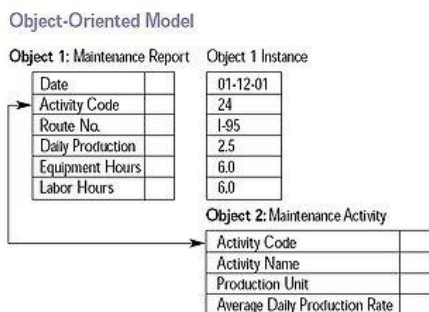
## Relacional



## Centralizada



## Orientado a Objetos



**Figura 7:** Modelos clásicos de tratamiento de los datos  
**Fuente:** <http://www.maestrosdelweb.com>  
**Elaborado por:** <http://www.maestrosdelweb.com>

La arquitectura de un sistema de base de datos se basa en 3 niveles distintos:

### Nivel Físico

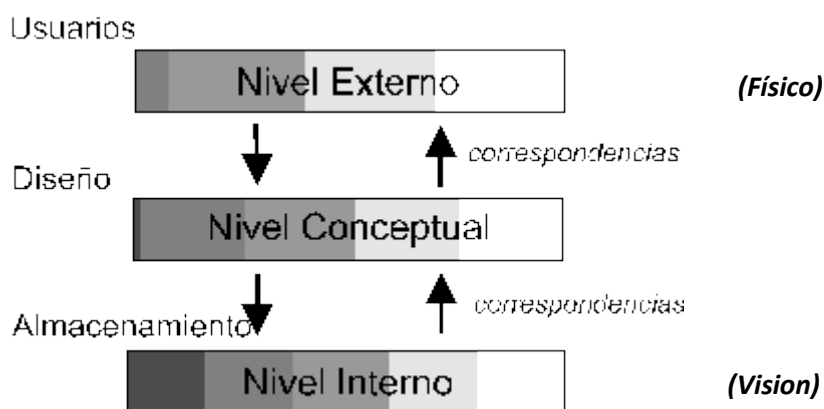
Es el nivel más bajo de abstracción y el nivel real de los datos almacenados. Este nivel define cómo se almacenan los datos en el soporte físico, ya sea en registros o de cualquier otra forma, así como los métodos de acceso. Este nivel lleva asociada una representación de los datos, que es lo que denominamos Esquema Físico.

### Nivel Conceptual

Es el correspondiente a una visión de la base de datos desde el punto de visto del mundo real. Es decir se trata con la entidad u objeto representado, sin importar como está representado o almacenado éste. Es la representación de los datos realizada por la organización, que recoge los datos parciales de los requerimientos de los diferentes usuarios y aplicaciones parciales. Incluye la definición de los datos y las relaciones entre ellos. Este nivel lleva asociado un Esquema Conceptual.

### ***Nivel de Visión***

Son partes del esquema conceptual. El nivel conceptual presenta toda la base de datos, mientras que los usuarios, por lo general, sólo tienen acceso a pequeñas parcelas de ésta. El nivel de visión es el encargado de dividir estas parcelas. Un ejemplo sería el caso del empleado de una organización que tiene acceso a la visión de su nómina, pero no a la de sus compañeros. El esquema asociado a éste nivel es el Esquema de Visión.



El modelo de datos es una cuestión fundamental a la hora de diseñar bases de datos. Existen estos 3 modelos fundamentales:

### ***Modelos lógicos basados en objetos***

Los dos más extendidos son el modelo entidad-relación y el orientado a objetos. El modelo entidad-relación (E-R) se basa en una percepción del mundo compuesta por objetos, llamados entidades, y relaciones entre ellos. Las entidades se diferencian unas de otras a través de atributos. El orientado a objetos también se basa en objetos, los cuales contienen valores y métodos, entendidos como órdenes que actúan sobre los valores, en niveles de anidamiento. Los objetos se agrupan en clases, relacionándose mediante el envío de mensajes. Algunos autores definen estos modelos como "modelos semánticos".

### ***Modelos lógicos basados en registros***

El más extendido es el relacional, mientras que los otros dos existentes, jerárquico y de red, se encuentran en retroceso. Estos modelos se usan para especificar la estructura lógica global de la base de datos, estructurada en registros de formato fijo de varios tipos. El

modelo relacional representa los datos y sus relaciones mediante tablas bidimensionales, que contienen datos tomados de los dominios correspondientes. El modelo de red está formado por colecciones de registros, relacionados mediante punteros o ligas en grafos arbitrarios. El modelo jerárquico es similar al de red, pero los registros se organizan como colecciones de árboles. Algunos autores definen estos modelos como "modelos de datos clásicos".

### *Modelos físicos de datos*

Muy poco usados, son el modelo unificador y el de memoria de elementos. Algunos autores definen estos modelos como "modelos de datos primitivos".

## 2.2.2 Lenguajes de Programación



Un lenguaje de programación es un lenguaje diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo.

Por otro lado, el término "lenguaje natural" define un medio de comunicación compartido por un grupo de personas (por ejemplo: inglés o francés).

Los lenguajes que los equipos usan para comunicarse entre ellos no tienen nada que ver con los lenguajes de programación; se los conoce como protocolos de comunicación. Se trata de dos conceptos totalmente diferentes. Un lenguaje de programación es muy estricto:

***“A CADA instrucción le corresponde UNA acción de procesador.”***

El lenguaje utilizado por el procesador se denomina lenguaje máquina. Se trata de datos tal como llegan al procesador, que consisten en una serie de 0 y 1 (datos binarios).

El lenguaje máquina, por lo tanto, no es comprensible para los seres humanos, razón por la cual se han desarrollado lenguajes intermediarios comprensibles para el hombre. El código escrito en este tipo de lenguaje se transforma en código máquina para que el procesador pueda procesarlo.

El ensamblador fue el primer lenguaje de programación utilizado. Es muy similar al lenguaje máquina, pero los desarrolladores pueden comprenderlo. No obstante, este lenguaje se parece tanto al lenguaje máquina que depende estrictamente del tipo de procesador utilizado (cada tipo de procesador puede tener su propio lenguaje máquina). Así, un programa desarrollado para un equipo no puede ser portado a otro tipo de equipo. El término "portabilidad" describe la capacidad de usar un programa de software en diferentes tipos de equipos. Para poder utilizar un programa de software escrito en un código ensamblador en otro tipo de equipo, ¡a veces será necesario volver a escribir todo el programa!

```
010000000010100010110000001001010001
110001011010001000111111111010000010
010100101100001010110110101011001000
1101100000101010001000100011000100111
01001100101101001101100111011101111
00010101#include <stdio.h>010100001101
10010011int main()0101000111
10001001int main()01010111
10101001printf("Hello World");0001100
11100110return 42;0101011011
010000010100011000110100001101
0001100010001100011000110100001101
1001001101110101110111000000101000111
10001001000101011001001101110100010111
1010100111001010111000101010001100
11100110000101111101010011111000110
0100001111101000100101010111011
```

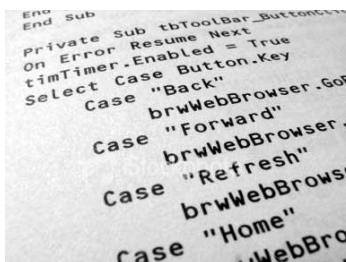
Por lo tanto, un lenguaje de programación tiene varias ventajas:

- Es mucho más fácil de comprender que un lenguaje máquina
- Permite mayor portabilidad, es decir que puede adaptarse fácilmente para ejecutarse en diferentes tipos de equipos.

### ***Grupos de Lenguajes de Programación***

Los lenguajes de programación generalmente se dividen en dos grupos principales en base al procesamiento de sus comandos:

#### ***Lenguaje de programación imperativo***



Un lenguaje imperativo programa mediante una serie de comandos, agrupados en bloques y compuestos de órdenes condicionales que permiten al programa retornar a un bloque de comandos si se cumple la condición. Estos fueron los primeros lenguajes de programación en uso y aún hoy muchos lenguajes modernos usan este principio.

No obstante, los lenguajes imperativos estructurados carecen de flexibilidad debido a la secuencialidad de las instrucciones.

## *Lenguaje de programación funcional*

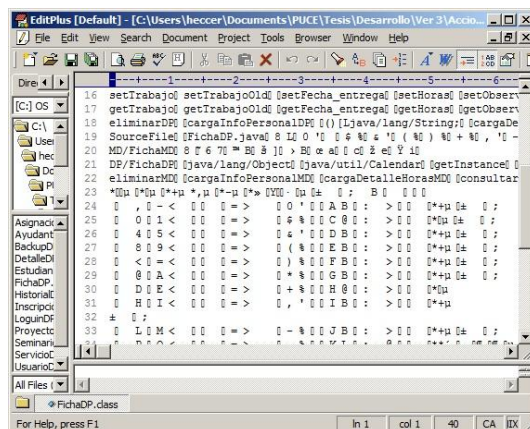
A menudo llamado lenguaje procedimental, es un lenguaje que crea programas mediante funciones, devuelve un nuevo estado de resultado y recibe como entrada el resultado de otras funciones. Cuando una función se invoca a sí misma, hablamos de recursividad.



Los lenguajes de programación pueden, en líneas generales, dividirse en dos categorías:

### *Lenguaje interpretado*

Un lenguaje de programación es, por definición, diferente al lenguaje máquina. Por lo tanto, debe traducirse para que el procesador pueda comprenderlo. Un programa escrito en un lenguaje interpretado requiere de un programa auxiliar (el intérprete), que traduce los comandos de los programas según sea necesario.



## Lenguaje compilado

Un programa escrito en un lenguaje "compilado" se traduce a través de un programa anexo llamado compilador que, a su vez, crea un nuevo archivo independiente que no necesita ningún otro programa para ejecutarse a sí mismo. Este archivo se llama ejecutable.

Un programa escrito en un lenguaje compilado posee la ventaja de no necesitar un programa anexo para ser ejecutado una vez que ha sido compilado. Además, como sólo es necesaria una traducción, la ejecución se vuelve más rápida.

Sin embargo, no es tan flexible como un programa escrito en lenguaje interpretado, ya que cada modificación del archivo fuente (el archivo comprensible para los seres humanos: el archivo a compilar) requiere de la compilación del programa para aplicar los cambios.

Por otra parte, un programa compilado tiene la ventaja de garantizar la seguridad del código fuente. En efecto, el lenguaje interpretado, al ser directamente un lenguaje legible, hace que cualquier persona pueda conocer los secretos de fabricación de un programa y, de ese modo, copiar su código o incluso modificarlo. Por lo tanto, existe el riesgo de que los derechos de autor no sean respetados. Por otro lado, ciertas aplicaciones aseguradas necesitan confidencialidad de código para evitar las copias ilegales (transacciones bancarias, pagos en línea, comunicaciones seguras...).

### *Lenguajes intermediarios*



Algunos lenguajes pertenecen a ambas categorías (LISP, Java, Python...) dado que el programa escrito en estos lenguajes puede, en ciertos casos, sufrir una fase de compilación intermedia, en un archivo escrito en un lenguaje ininteligible (por lo tanto diferente al archivo fuente) y no ejecutable (requeriría un intérprete). Los applets Java, pequeños programas que a menudo se cargan en páginas web, son archivos compilados que sólo pueden ejecutarse dentro de un navegador web (son archivos con la extensión .class).

En la Tabla 1, se presenta una breve lista de los lenguajes de programación actuales:

Lenguaje	Principal área de aplicación	Compilado/interpretado
ADA	Tiempo real	Lenguaje compilado
BASIC	Programación para fines educativos	Lenguaje interpretado
C	Programación de sistema	Lenguaje compilado
C++	Programación de sistema orientado a objeto	Lenguaje compilado
Cobol	Administración	Lenguaje compilado
Fortran	Cálculo	Lenguaje compilado
Java	Programación orientada a Internet	Lenguaje intermedio
MATLAB	Cálculos matemáticos	Lenguaje interpretado
LISP	Inteligencia artificial	Lenguaje intermedio
Pascal	Educación	Lenguaje compilado
PHP	Desarrollo de sitios web dinámicos	Lenguaje interpretado
Perl	Procesamiento de cadenas de caracteres	Lenguaje interpretado

Tabla 1: Lenguajes actuales de programación y su tipo

Fuente: <http://frt.utn.edu.ar>

Elaborado por: <http://frt.utn.edu.ar>



### 2.2.3 Herramientas CASE



Se puede definir a las Herramientas CASE (Computer Aided Software Engineering) como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un Software. Como es sabido, los estados en el Ciclo de

Vida de desarrollo de un Software son: Investigación Preliminar, Análisis, Diseño, Implementación e Instalación.

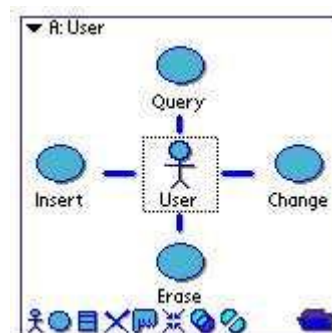
CASE se define también como un conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de sistemas de información, completamente o en alguna de sus fases.

La realización de un nuevo software requiere que las tareas sean organizadas y completadas en forma correcta y eficiente. Las Herramientas CASE fueron desarrolladas para automatizar esos procesos y facilitar las tareas de coordinación de los eventos que necesitan ser mejorados en el ciclo de desarrollo de software.

También permite a las compañías competir más efectivamente usando estos sistemas desarrollados nuevamente para compararlos con sus necesidades de negocio actuales.

En un mercado altamente competitivo, esto puede hacer la diferencia entre el éxito y el fracaso.

Las herramientas CASE también permiten a los analistas tener más tiempo para el análisis y diseño y minimizar el tiempo para codificar y probar.



La introducción de CASE integradas está comenzando a tener un impacto significativo en los negocios y sistemas de información de las organizaciones.

Con un CASE integrado, las organizaciones pueden desarrollar rápidamente sistemas de mejor calidad para soportar procesos críticos del negocio y asistir en el desarrollo y promoción intensiva de la información de productos y servicios.



Estas herramientas pueden proveer muchos beneficios en todas las etapas del proceso de desarrollo de software, algunas de ellas son:

- Verificar el uso de todos los elementos en el sistema diseñado.
- Automatizar el dibujo de diagramas.
- Ayudar en la documentación del sistema.
- Ayudar en la creación de relaciones en la Base de Datos.
- Generar estructuras de código.

La principal ventaja de la utilización de una herramienta CASE, es la mejora de la calidad de los desarrollos realizados y, en segundo término, el aumento de la productividad. Para conseguir estos dos objetivos es conveniente contar con una organización y una metodología de trabajo, además de la propia herramienta.

### ***Tipos de Herramientas CASE***

Las herramientas CASE, en función de las fases del ciclo de vida abarcadas, se pueden agrupar de la forma siguiente:

#### ***Herramientas integradas***

Llamadas también I-CASE (Integrated CASE, CASE integrado) abarcan todas las fases del ciclo de vida del desarrollo de sistemas.

Son llamadas también CASE workbench.

#### ***Herramientas de alto nivel***

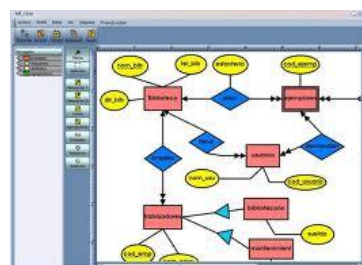
Llamadas también U-CASE (Upper CASE - CASE superior) o front-end, orientadas a la automatización y soporte de las actividades desarrolladas durante las primeras fases del desarrollo: análisis y diseño.

#### ***Herramientas de bajo nivel***

Llamadas también L-CASE (Lower CASE - CASE inferior) o back-end, dirigidas a las últimas fases del desarrollo: construcción e implantación.

#### ***Juegos de herramientas***

Llamadas también Tools-Case, son el tipo más simple de herramientas CASE. Automatizan una fase dentro del ciclo de vida.



## IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

Dentro de este grupo se encontrarían las herramientas de reingeniería, orientadas a la fase de mantenimiento.

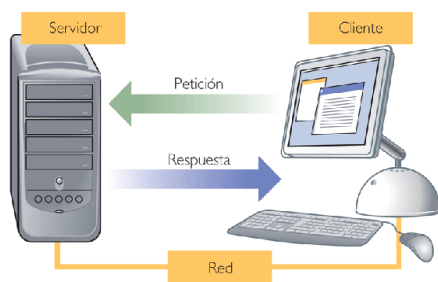
En la Tabla 2, se muestran las principales ventajas y desventajas de cada tipo de CASE:

Tipo de Case	Ventajas	Desventajas
<b>I – Case</b>	<ul style="list-style-type: none"> <li>➤ Integra el ciclo de vida.</li> <li>➤ Permite lograr importantes mejoras de productividad a mediano plazo.</li> <li>➤ Permite un eficiente soporte al mantenimiento de sistemas.</li> <li>➤ Mantiene la consistencia de los sistemas a nivel corporativo.</li> </ul>	<ul style="list-style-type: none"> <li>➤ No es tan eficiente para soluciones simples, sino para soluciones complejas.</li> <li>➤ Depende del Hardware y del Software.</li> <li>➤ Es costoso.</li> </ul>
<b>Upper Case</b>	<ul style="list-style-type: none"> <li>➤ Se utiliza en plataforma PC, es aplicable a diferentes entornos,</li> <li>➤ Menor costo</li> </ul>	<ul style="list-style-type: none"> <li>➤ Permite mejorar la calidad de los sistemas, pero no mejora la productividad.</li> <li>➤ No permite la integración del ciclo de vida.</li> </ul>
<b>Lower Case</b>	<ul style="list-style-type: none"> <li>➤ Permite lograr importantes mejoras de productividad a corto plazo.</li> <li>➤ Permite un eficiente soporte al mantenimiento de sistemas.</li> </ul>	<ul style="list-style-type: none"> <li>➤ No garantiza la consistencia de los resultados a nivel corporativo.</li> <li>➤ No garantiza la eficiencia del Análisis y Diseño.</li> <li>➤ No permite la integración del ciclo de vida.</li> </ul>

**Tabla 2:** Lenguajes actuales de programación y su tipo  
**Fuente:** INSTITUTO PERUANO DE ESTADISTICAS E INFORMATICA. HERRAMIENTAS CASE.  
**Elaborado por:** <http://www.inei.gob.pe>

## 2.3 Arquitectura de Aplicaciones

### 2.3.1 Arquitectura Cliente/Servidor



La tecnología Cliente/Servidor es el procesamiento cooperativo de la información por medio de un conjunto de procesadores, en el cual múltiples clientes, distribuidos geográficamente, solicitan requerimientos a uno o más servidores centrales.

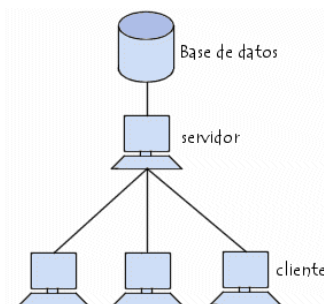
Desde el punto de vista funcional, se puede definir la computación Cliente/Servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la

información de forma transparente aún en entornos multiplataforma. Se trata pues, de la arquitectura más extendida en la realización de Sistemas de información distribuido.

*Un sistema Cliente/Servidor es un Sistema de Información distribuido basado en las siguientes características:*

- **Servicio:** unidad básica de diseño. El servidor los proporciona y el cliente los utiliza.
- **Recursos compartidos:** Muchos clientes utilizan los mismos servidores y, a través de ellos, comparten tanto recursos lógicos como físicos.
- **Protocolos asimétricos:** Los clientes inician “conversaciones”. Los servidores esperan su establecimiento pasivamente.
- **Transparencia de localización física de los servidores y clientes:** El cliente no tiene por qué saber dónde se encuentra situado el recurso que desea utilizar.
- **Encapsulamiento de servicios:** Los detalles de la implementación de un servicio son transparentes al cliente.

Para entender en forma más ordenada y clara los conceptos y elementos involucrados en esta tecnología se puede aplicar una descomposición o arquitectura de niveles. Esta descomposición principalmente consiste en separar los elementos estructurales de esta tecnología en función de aspectos más funcionales de la misma:



- **Nivel de Presentación:** Agrupa a todos los elementos asociados al componente Cliente.
- **Nivel de Aplicación:** Agrupa a todos los elementos asociados al componente Servidor.
- **Nivel de Comunicación:** Agrupa a todos los elementos que hacen posible la comunicación entre los componentes Cliente y servidor.
- **Nivel de Base de Datos:** Agrupa a todas las actividades asociadas al acceso de los datos.

### *Elementos principales*



#### **Cliente**

Un cliente es todo proceso que reclama servicios de otro. Una definición un poco más elaborada podría ser la siguiente: cliente es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor. Se lo conoce con el término front-end.



#### **Servidor**

Un servidor es todo proceso que proporciona un servicio a otros. Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se lo conoce con el término back-end. El servidor normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos.



Petición

Respuesta



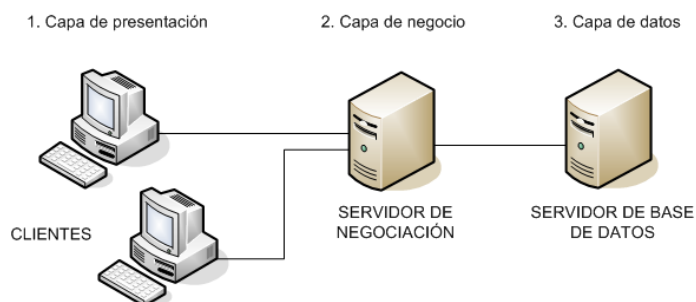
Red

#### **Middleware**

El middleware es un módulo intermedio que actúa como conductor entre sistemas permitiendo a cualquier usuario de sistemas de información comunicarse con varias fuentes de información que se encuentran conectadas por una red. En el caso que nos concierne, es el intermediario entre el cliente y el servidor y se ejecuta en ambas partes.

Dentro de los inconvenientes más importantes destacan la mayor carga de máquina necesaria para que puedan funcionar.

### **2.3.2 Arquitectura por 3 Capas**



La programación por capas es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto

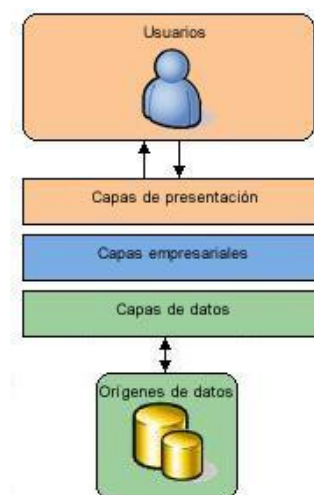
consiste en separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Un buen ejemplo de este método de programación sería el modelo de interconexión de sistemas abiertos.

En el diseño de sistemas informáticos actuales, se suelen usar las arquitecturas multinivel o programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

*El diseño más utilizado actualmente es el diseño en tres niveles (o en tres capas).*

La arquitectura 3 capas o programación 3 capas consiste literalmente en separar un proyecto en Capa de Presentación, Capa de Negocio y Capa de Datos. Esto permite distribuir el trabajo de creación de una aplicación por niveles; de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles.



### ***Ventajas***

- El desarrollo se puede llevar a cabo en varios niveles.
- Desarrollos paralelos (en cada capa).
- Aplicaciones más robustas debido al encapsulamiento.
- En caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.
- Mantenimiento y soporte más sencillo (es más sencillo cambiar un componente que modificar una aplicación monolítica).
- Mayor flexibilidad (se pueden añadir nuevos módulos para dotar al sistema de nueva funcionalidad).

- Alta escalabilidad. La principal ventaja de una aplicación distribuida bien diseñada es alta escalabilidad, es decir, que puede manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware.

### *Capas y niveles*

#### *Capa de Presentación*



Esta es la parte que ve el usuario, las pantallas que se le muestra para que el interactúe con el programa (también se le conoce como “capa de usuario”), comunicándole la información y recolectando la información suministrada por el usuario en un mínimo de proceso (realiza validaciones para comprobar que no hay errores de formato).

#### *Capa de negocio*

Es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todos los procesos que deben realizarse.



#### *Capa de datos*



Es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

La selección de las técnicas y herramientas referenciadas en este capítulo, se detalla en la sección 3.3. Justificación de las herramientas seleccionadas para la construcción de la aplicación.

## CAPITULO 3: CASO DE ESTUDIO

### Proceso de gestión de horas de Acción Social Universitaria de la Facultad de Ingeniería de la PUCE

#### 3.1 Introducción

La Facultad de Ingeniería de la PUCE, requiere un sistema que permita manejar la información de actividades realizadas por los estudiantes durante toda su carrera universitaria. Para ello, se acordó desarrollar un sistema que administre las entidades necesarias para poder manejar los procesos del área. Entre los puntos tratados con el área, se concluyó que el sistema permitirá manejar las siguientes actividades.

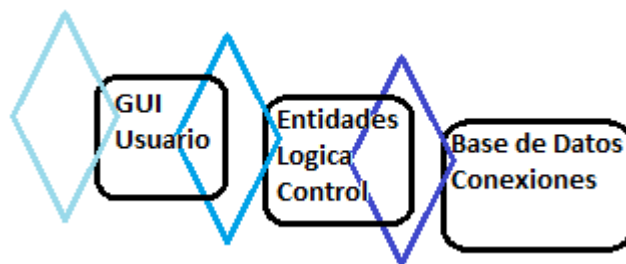
#### 3.2 Principales Actividades

- Administración de entidades usadas:
  - Ayudantías
  - Estudiantes
  - Proyectos
  - Seminarios
  - Servicios
  - Usuarios
- Manejo de procesos críticos:
  - Gestión de horas por estudiante
- Manejo de procesos no críticos:
  - Reportes de actividades por estudiante
  - Impresión de ficha de horas por estudiante
  - Respaldo de información en cualquier instante

#### 3.3 Justificación de las herramientas seleccionadas para la construcción de la aplicación

##### 3.3.1 Metodología

Se aplicará una metodología de investigación descriptiva, ya que se requerirá de un análisis previo del problema, y describir las características del producto a partir de sus requerimientos. Además se utilizará un **modelo orientado a objetos con tres capas**.



### 3.3.2 Diseño

#### Herramientas



manera flexible, rápida y amigable.

Para el diseño de la aplicación se usará una herramienta UML libre llamada **StarUML**. Esta herramienta permite crear diagramas UML de

El uso de UML para diseñar aplicaciones, da la facilidad de tener una visión más elaborada de lo que queremos desarrollar específicamente, sin dejar procesos a medias.

La idea principal, es diseñar todos los comportamientos, casos de uso, clases, secuencias de una manera estructurada y consistente para tomar mucho menos tiempo en el desarrollo de la aplicación.

### 3.3.3 Desarrollo

#### Técnicas

Se tomará la técnica de desarrollo **RAD** (Diseño Rápido de Aplicaciones) ya que el tiempo para entregar el producto es corto, y los requisitos y fronteras del software son limitados. Esta técnica es buena, pues permite la interacción directa con el cliente final y la utilización de prototipos para agilizar el desarrollo. La necesidad de un grupo de desarrolladores es dependiente de la complejidad del producto, por lo cual, para este caso es necesario un solo desarrollador.

La técnica de desarrollo RAD se basa en el ciclo de vida general del desarrollo de software, tomando en cuenta que se trabaja con prototipos y con la interacción cara a cara con el usuario, se define el ciclo de la siguiente manera:



*Inicio*

Análisis de requerimientos

Especificaciones de funcionalidad

Arquitectura

Diseño

*Elaboración*

Prototipo ejecutable

Lista de riesgos

*Construcción*

Implementación

Pruebas

*Transición*

Seguridades

Despliegue

Mantenimiento

***Herramientas***

Dado que el sistema será usado por personal docente de la facultad de Ingeniería de la P.U.C.E, este tiene que ser necesariamente elaborado en Software Libre, ya que es uno de los estándares de software interno desarrollado en la Facultad.



Se utilizará **MySQL Server 5.1**, que es la misma base de datos que usa la Facultad para el manejo de sus estudiantes. La integración de la aplicación con esta base de datos, es para aprovechar la información de los estudiantes y poder utilizarla en la presentación de su información en las fichas.

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

El sistema será desarrollado en lenguaje **Java**, que es un lenguaje de programación muy robusto, rápido y lo más importante, libre. Bajo los requerimientos del cliente, será desarrollado como aplicación de escritorio, con un ambiente de ventanas internas.



Se inclinó por este lenguaje de programación, ya que ha sido la herramienta de desarrollo que la facultad se ha esforzado en usar más. Por este motivo, usaran todas las técnicas de desarrollo aprendidas en la carrera como Ingeniería de Software.

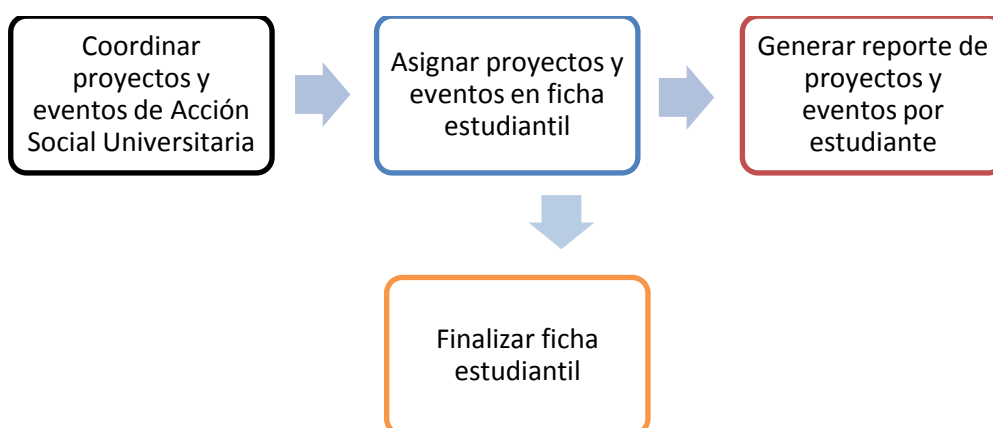
Además, su uso se respalda por la metodología de desarrollo usada (RAD), ya que recomienda usar prototipos reusables, para lo cual, se reusará un sistema de gestión de entidades básico presentado en los últimos niveles de la carrera como proyecto final, y que está desarrollado en Java.

## CAPITULO 4: LEVANTAMIENTO DEL PROCESO

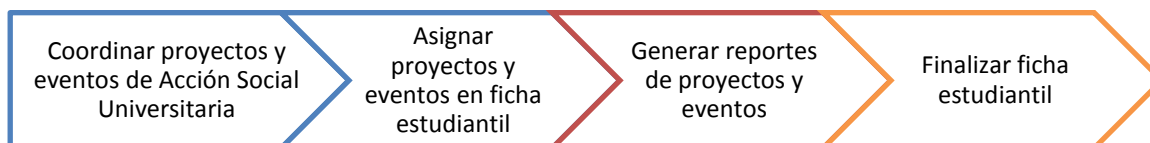
### 4.1 Mapa de Procesos

El área de Acción Social Universitaria de la Facultad de Ingeniería de la PUCE, se encarga entre otras cosas, de la administración de horas que los estudiantes manejan en los diferentes proyectos durante toda su carrera. Los procesos más resaltantes del área, basándose en la implementación del Sistema de Gestión de horas de Acción Social Universitaria, son:

- Coordinar proyectos y eventos de Acción Social Universitaria
- Asignar proyectos y eventos en ficha estudiantil
- Generar reporte de proyectos y eventos por estudiante
- Finalizar ficha estudiantil



#### 4.1.1 Cadena de Valor



### 4.2 SIPOC

#### 4.2.1 Proceso

Asignar proyectos y eventos en ficha estudiantil

#### 4.2.2 Diagrama

Proveedores	Entrada	Proceso	Salida	Clientes
<ul style="list-style-type: none"> <li>Estudiantes</li> <li>Proyectos y Eventos</li> </ul>	<ul style="list-style-type: none"> <li>Información Estudiante</li> <li>Información Proyecto</li> <li>Información Evento</li> <li>Horas acción social</li> <li>Fechas de servicio</li> </ul>	<pre> graph TD     Inicio([Inicio]) --&gt; Obtener1[/Obtener información/]     Obtener1 --&gt; Existe1{Existe estudiante?}     Existe1 -- No --&gt; Crear1[Crear estudiante]     Crear1 --&gt; Obtener1     Existe1 -- Si --&gt; Obtener2[/Obtener información/]     Obtener2 --&gt; Existe2{Existe evento?}     Existe2 -- No --&gt; Crear2[Crear evento]     Crear2 --&gt; Asignar1[Asignar fechas de servicio]     Existe2 -- Si --&gt; Asignar1     Asignar1 --&gt; Asignar2[Asignar horas de acción social]     Asignar2 --&gt; Guardar[Guardar Ficha]     Guardar --&gt; Fin([Fin])     Guardar -- "Ficha horas acción social" --&gt; Asignar2     </pre>	<ul style="list-style-type: none"> <li>Reporte de Proyectos y Eventos por estudiante.</li> <li>Almacenamiento de Proyectos y Eventos de acción social del estudiante.</li> <li>Información almacenada de nuevos estudiantes.</li> <li>Información almacenada de nuevos Proyectos y Eventos.</li> </ul>	<ul style="list-style-type: none"> <li>Estudiantes</li> </ul>

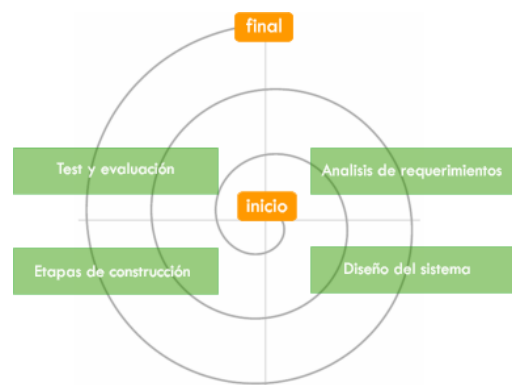
## CAPITULO 5: CONSTRUCCION DEL PRODUCTO

Para la elaboración del producto se usará el ciclo de vida básico RAD, envuelto en las 4 fases básicas de desarrollo de aplicaciones.

- Inicio
- Elaboración
- Construcción
- Transición

Donde, en la fase **inicial**, se recolectan los requerimientos de funcionalidad de la aplicación. Luego, con la suficiente información obtenida, se generan los diagramas UML que muestran la arquitectura inicial de la aplicación antes de elaborarla. En esta fase también se muestran las imágenes (diseño) de cada ventana mostrada en la aplicación.

En la fase de **elaboración** se presentará un prototipo básico ejecutable que mostrará las funcionalidades que tendrá el producto luego de su construcción, además se filtrará la lista de posibles riesgos en el transcurso del ciclo de vida.



En la fase de **construcción** se hará uso de las técnicas de desarrollo de software aprendidas durante la carrera, implementando interfaces fácilmente reproducibles y herencias. Bajo cada módulo construido se procederá a hacer las respectivas pruebas de funcionalidad y diseño. Aquí se almacenarán los requerimientos faltantes del ciclo y se sumarán al siguiente ciclo.

Con esto se acaba el primer ciclo de construcción, preparándose para el siguiente análisis de requerimientos y elaboración de los diagramas UML respectivos al caso.

En la fase de **transición**, se asume que los ciclos de vida de RAD se han completado exitosamente, para luego preocuparnos por detalles finales durante la entrega del producto, como la seguridad de los datos y accesos, costos de construcción, instalación y mantenimiento, y finalmente, despliegue y entrega del producto al cliente final.

## 5.1 Inicio

### 5.1.1 *Análisis De Requerimientos*

Es importante iniciar con un acuerdo preciso de lo que es el producto y así tener un mecanismo que proteja los requerimientos originales de los desacuerdos encontrados; eso será posible con el documento de requerimientos claro y preciso, con el cual se podrán administrar los cambios y proteger tal documento contra un cliente que insista en interpretar de manera diferente una función que ya ha sido diseñada.

En el documento de requerimientos se describe la interpretación de la definición de necesidades; después de que los clientes hayan leído ese documento y estén de acuerdo en su contenido se podrá argumentar que cualquier cambio costará tiempo y/o dinero, en esto radica la importancia de dicho documento.

#### *Objetivo*

Con frecuencia se suele hacer cambios a los requerimientos, generalmente, los usuarios no pueden saber lo que necesitan en forma precisa hasta que tratan de usar el producto terminado. Sin embargo, cuando se introduce un sistema nuevo se cambia la manera en que ellos trabajan. Conforme el trabajo de desarrollo progresa empiezan a apreciar la manera en cómo afectará su ambiente, mientras más claro es eso para los usuarios, pensarán en nuevas funciones y características, por ésto, los requerimientos constantemente cambian hasta que se congelan en un producto.

La parte difícil del proceso de requerimientos de software es entender que es lo que los usuarios creen que necesitan y ayudarlos a definir esas necesidades en términos de funciones que generen un producto útil. El siguiente paso es realizar la SRS que represente el acuerdo común entre los usuarios y el equipo de trabajo acerca de lo que necesitan.

Es importante iniciar con un acuerdo preciso de lo que es el producto y así tener un mecanismo que proteja los requerimientos originales de los desacuerdos encontrados; eso será posible con un documento de requerimientos claro y preciso, con la cual se podrán administrar los cambios y proteger tal documento contra un cliente que insista en interpretar de manera diferente una función que ya ha sido diseñada.

### ***Definición General del Sistema***

#### ***Antecedentes***

**Tema:** Sistema de Gestión de horas de Extensión Universitaria para la Facultad de Ingeniería de la PUCE

Se ha visto necesario el desarrollo del presente sistema, puesto que en la actualidad el manejo de esta información no está automatizada.

#### ***Estrategia***

Consiste en aplicar un proceso cíclico para desarrollar el producto. En cada ciclo se deciden cuáles funcionalidades desarrollar. En el primero de ellos se diseña, implementa y evalúa una primera versión del producto; en el segundo ciclo, se incrementan las funcionalidades del producto para generar una segunda versión. Si hay tiempo, se produce una tercera versión.

### ***Requerimientos Funcionales***

**F0.** El usuario podrá ingresar al sistema con sus credenciales

**F1.** El usuario podrá administrar información de los Proyectos

**F1.1.** El usuario podrá ingresar Proyectos

**F1.2.** El usuario podrá modificar Proyectos

**F1.3.** El usuario podrá eliminar Proyectos

**F2.** El usuario podrá administrar información de las Ayudantías

**F2.1.** El usuario podrá ingresar Ayudantías

**F2.2.** El usuario podrá modificar Ayudantías

**F2.3.** El usuario podrá eliminar Ayudantías

**F3.** El usuario podrá administrar información de los Seminarios

**F3.1.** El usuario podrá ingresar Seminarios

**F3.2.** El usuario podrá modificar Seminarios

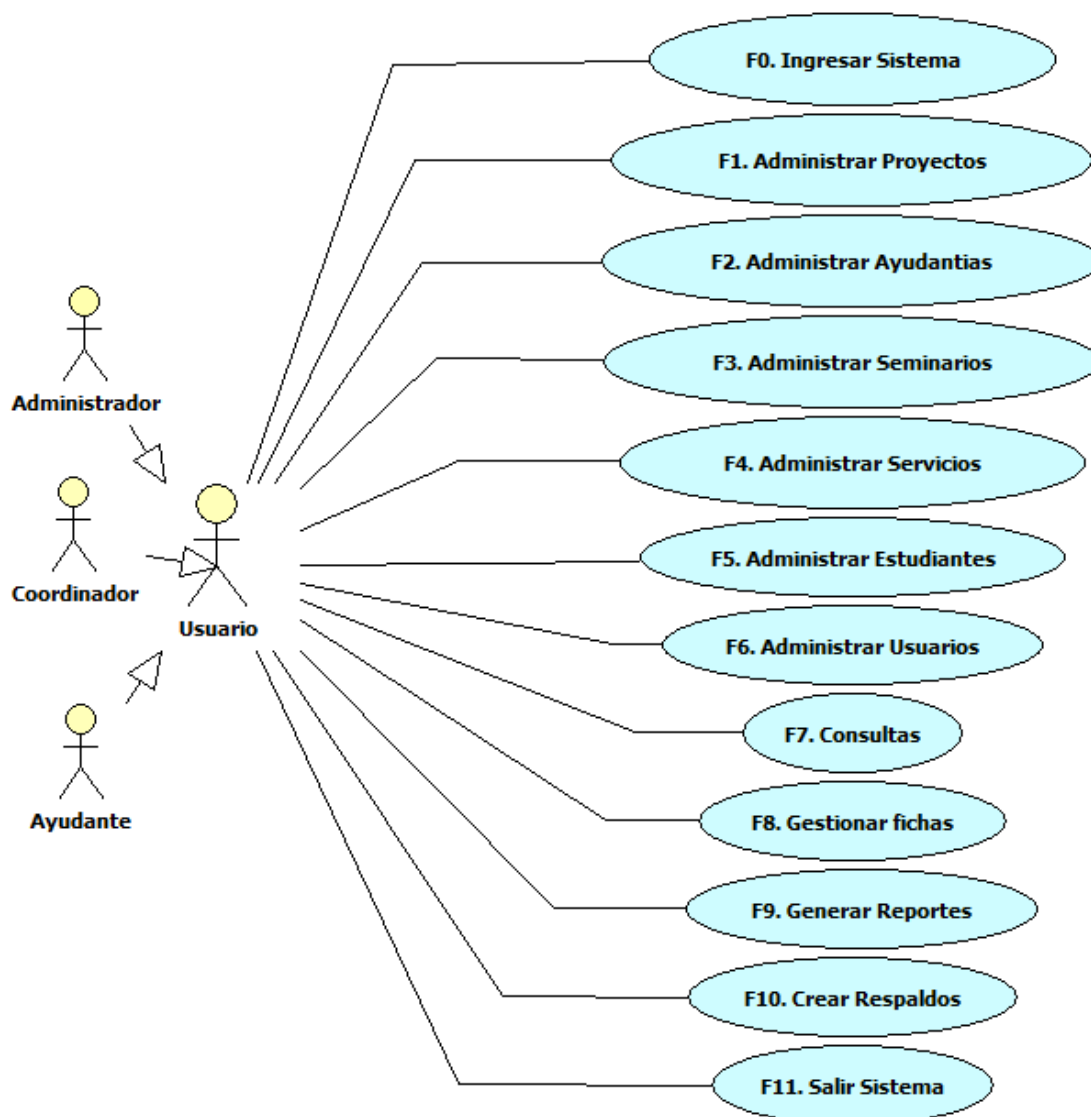
**F3.3.** El usuario podrá eliminar Seminarios

- F4.** El usuario podrá administrar información de los Servicios
  - F4.1.** El usuario podrá ingresar Servicios
  - F4.2.** El usuario podrá modificar Servicios
  - F4.3.** El usuario podrá eliminar Servicios
- F5.** El usuario podrá administrar información de los Estudiantes
  - F5.1.** El usuario podrá ingresar Estudiantes
  - F5.2.** El usuario podrá modificar Estudiantes
  - F5.3.** El usuario podrá eliminar Estudiantes
- F6.** El usuario administrador podrá administrar los Usuarios del sistema
  - F6.1.** El usuario administrador podrá ingresar Usuarios
  - F6.2.** El usuario administrador podrá modificar Usuarios
  - F6.3.** El usuario administrador podrá eliminar Usuarios
- F7.** El usuario podrá Consultar la información almacenada en el Sistema
  - F7.1.** El usuario podrá consultar Proyectos
    - F7.1.1.** El usuario podrá consultar Proyectos en general
    - F7.1.2.** El usuario podrá consultar Proyectos por parámetros
  - F7.2.** El usuario podrá consultar Ayudantías
    - F7.2.1.** El usuario podrá consultar Ayudantías en general
    - F7.2.2.** El usuario podrá consultar Ayudantías por parámetros
  - F7.3.** El usuario podrá consultar Seminarios
    - F7.3.1.** El usuario podrá consultar Seminarios en general
    - F7.3.2.** El usuario podrá consultar Seminarios por parámetros
  - F7.4.** El usuario podrá consultar Servicios
    - F7.4.1.** El usuario podrá consultar Servicios en general
    - F7.4.2.** El usuario podrá consultar Servicios por parámetros
- F7.5.** El usuario podrá consultar Estudiantes



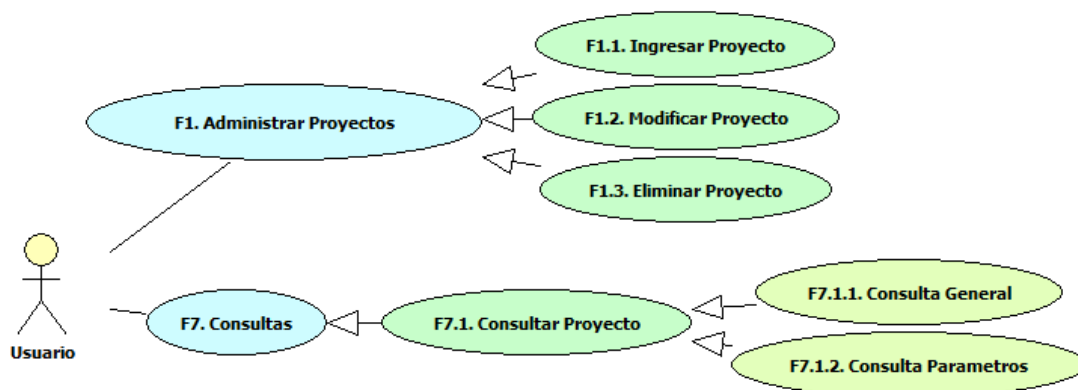
- F7.5.1.** El usuario podrá consultar Estudiantes en general
  - F7.5.2.** El usuario podrá consultar Estudiantes por parámetros
- F7.6.** El usuario administrador podrá consultar Usuarios
  - F7.6.1.** El usuario podrá consultar Usuarios en general
  - F7.6.2.** El usuario podrá consultar Usuarios por parámetros
- F7.7.** El usuario podrá consultar el Historial
  - F7.7.1.** El usuario podrá consultar el Historial en general
  - F7.7.2.** El usuario podrá consultar el Historial por parámetros
- F7.8.** El usuario podrá consultar Fichas Finalizadas
  - F7.8.1.** El usuario podrá consultar Fichas Finalizadas en general
  - F7.8.2.** El usuario podrá consultar Fichas Finalizadas por parámetros
- F8.** El usuario podrá gestionar las fichas de acción social del Estudiante
  - F8.1.** El usuario podrá administrar horas de trabajos en la Ficha
    - F8.1.1.** El usuario podrá ingresar horas de trabajos
    - F8.1.2.** El usuario podrá modificar horas de trabajos
    - F8.1.3.** El usuario administrador podrá eliminar horas de trabajos
  - F8.2.** El usuario podrá generar el reporte de la ficha de acción social
  - F8.3.** El usuario podrá finalizar la ficha del Estudiante para cerrarla
  - F8.4.** El usuario no podrá re-abrir las fichas finalizadas
- F9.** El usuario podrá generar reportes de fichas de actividades estudiantiles
  - F9.1.** El usuario podrá generar reportes de actividades por Trabajo
  - F9.1.** El usuario podrá generar reportes de actividades generales por fechas
- F10.** El usuario podrá crear respaldos de la información en unidades extraíbles
- F11.** El usuario podrá salir del sistema en cualquier momento

*Diagrama de Casos de Uso General (nivel 0)*



### 5.1.2 Especificaciones de Funcionalidad

#### F1. Administración de proyectos

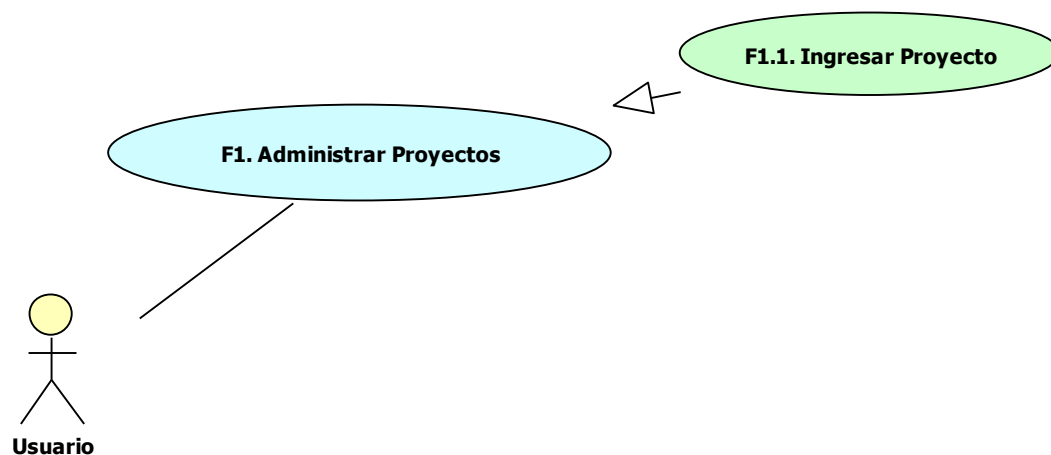


#### Detalle:

- F1.1. Ingreso de Proyectos**

*A través de este caso uso se ingresarán nuevos proyectos de Acción Social*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Proyectos	2	Mostrar ventana de Administración de Proyectos	

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

<b>3</b>	Ingresar el Identificador de proyecto en el TextBox inicial de la ventana.	<b>4</b>	Verificar Identificador del proyecto en la Base de Datos	E1
<b>5</b>	Ingresar el resto de datos en la ventana			
<b>6</b>	Presionar el botón GUARDAR para guardar los datos del proyecto.	<b>7</b>	Ingresar nuevo proyecto en la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

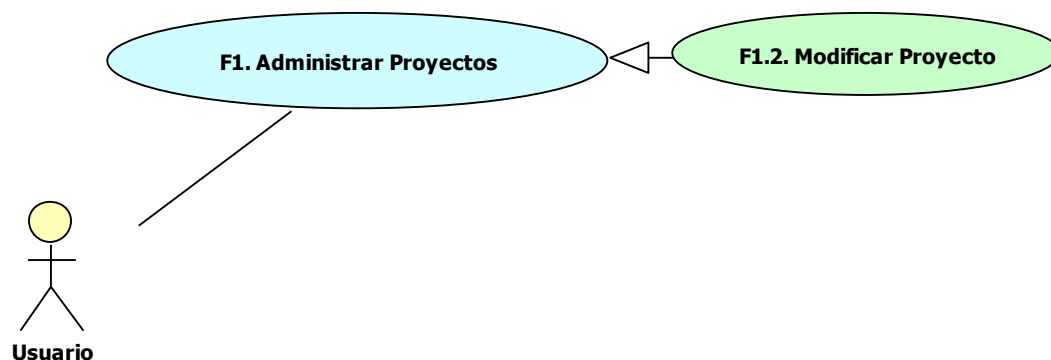
**Extensiones**

Paso	Descripción
4	Si existe, verificar F1.2 o F1.3

- F1.2. Modificación de Proyectos**

*A través de este caso de uso se modificarán todos los datos de un Proyecto.*

- Actores:** Usuario



**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Proyectos	2	Mostrar ventana de Administración de Proyectos	
3	Ingresar el Identificador del proyecto en el TextBox inicial de la ventana.	4	Verificar Identificador del proyecto en la base de datos	E1
		5	Mostrar datos del proyecto en los TextBox's restantes de la ventana	
6	Modificar datos del proyecto en la ventana			
7	Presionar el botón GUARDAR para guardar los datos del proyecto	8	Actualizar proyecto en la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

**Extensiones**

Paso	Descripción
4	Si no existe, verificar F1.1

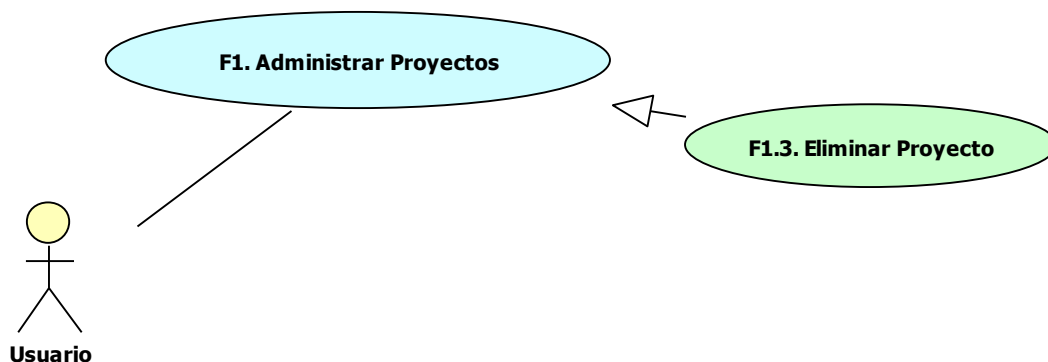
- F1.3. Eliminación de Proyectos**

*A través de este caso se eliminará un proyecto del Sistema.*



# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

- **Actores:** Usuario



- **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Proyectos	2	Mostrar ventana de Administración de Proyectos	
3	Ingresar el Identificador del proyecto en el TextBox inicial de la ventana.	4	Verificar Identificador del proyecto en la base de datos	E1
		5	Mostrar datos del proyecto en los TextBox Restantes de la ventana	
6	Presionar el botón ELIMINAR para borrar el proyecto	7	Eliminar proyecto de la base de datos.	E2

## Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Eliminar	Volver a ingresar el ID



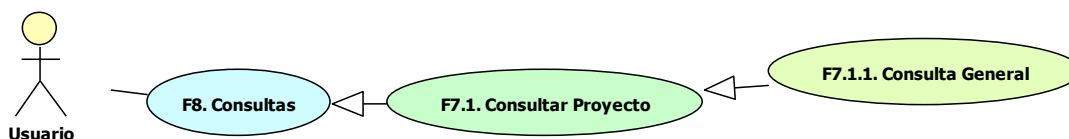
## Extensiones

Paso	Descripción
4	Si no existe, verificar F1.1

- F7.1.1. Consulta General de Proyectos**

*A través de este caso de uso se mostrarán todos los proyectos en el Sistema.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Proyectos, submenú General	2	Mostrar ventana con tabla de proyectos	E1

## Excepciones

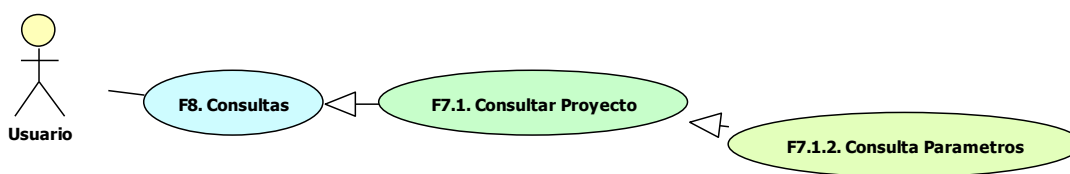
Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- F7.1.2. Consulta por Parámetros de Proyectos**

*A través de este caso de uso se mostrarán todos los proyectos en el Sistema según los parámetros ingresados.*

- Actores:** Usuario

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**



• **Descripción**

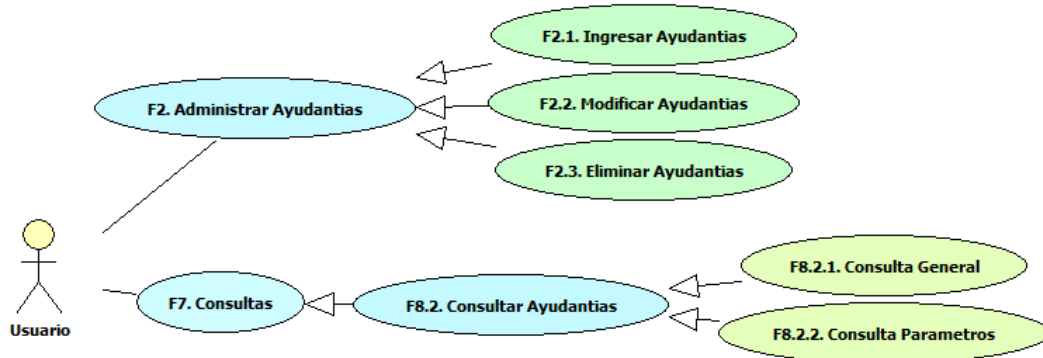
Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción proyectos, submenú parámetros	2	Mostrar ventana de Consulta por parámetros de proyectos	
3	Seleccionar parámetro en el ComboBox de la Ventana.			
4	Ingresar Valor del parámetro			
5	Presionar botón Consultar	6	Cargar datos de El proyectos en la tabla de la ventana	

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión



## F2. Administración de ayudantías

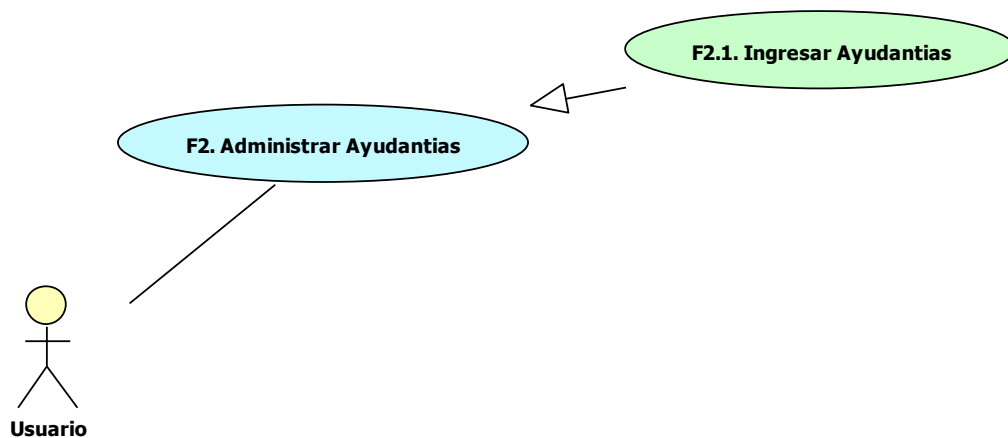


### Detalle:

- F2.1. Ingreso de Ayudantías**

*A través de este caso uso se ingresarán nuevas ayudantías de Acción Social*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Ayudantías	2	Mostrar ventana de Administración de Ayudantías	
3	Ingresar el Identificador de ayudantía en el TextBox	4	Verificar Identificador de la ayudantía en la	E1

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

	inicial de la ventana.	base de datos
<b>5</b>	Ingresar el resto de datos en la ventana	
<b>6</b>	Presionar el botón GUARDAR para guardar los datos de la ayudantía.	<b>7</b> Ingresar nueva ayudantía en la base de datos. E2

**Excepciones**

Código	Descripción	Alternativa
<b>E1</b>	No hay conexión a BDD	Reintentar conexión
<b>E2</b>	Problemas al Guardar	Pedir reingreso de datos

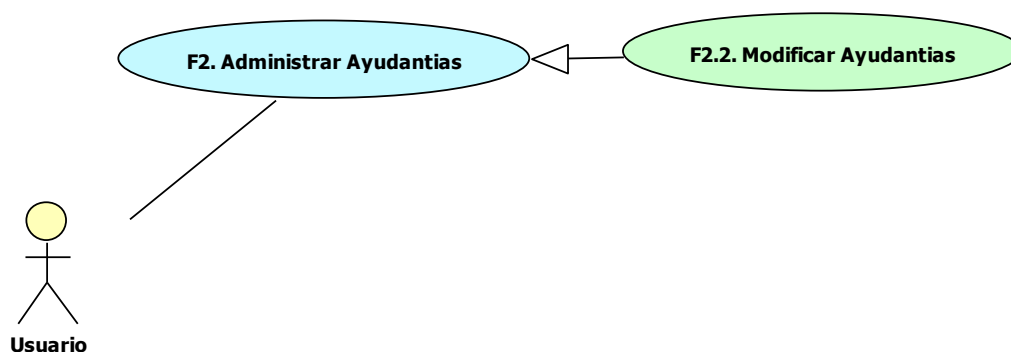
**Extensiones**

Paso	Descripción
<b>4</b>	Si existe, verificar F2.2 o F2.3

- F2.2. Modificación de Ayudantías**

*A través de este caso de uso se modificarán todos los datos de una ayudantía.*

- Actores:** Usuario



**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Ayudantías	2	Mostrar ventana de Administración de Ayudantías	
3	Ingresar el Identificador de la ayudantía en el TextBox inicial de la ventana.	4	Verificar identificador de la ayudantía en la base de datos	E1
		5	Mostrar datos de la ayudantía en los TextBox's restantes de la ventana	
6	Modificar datos de la ayudantía en la ventana			
7	Presionar el botón GUARDAR para guardar los datos de la ayudantía	8	Actualizar ayudantía en la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

**Extensiones**

Paso	Descripción
4	Si no existe, verificar F2.1

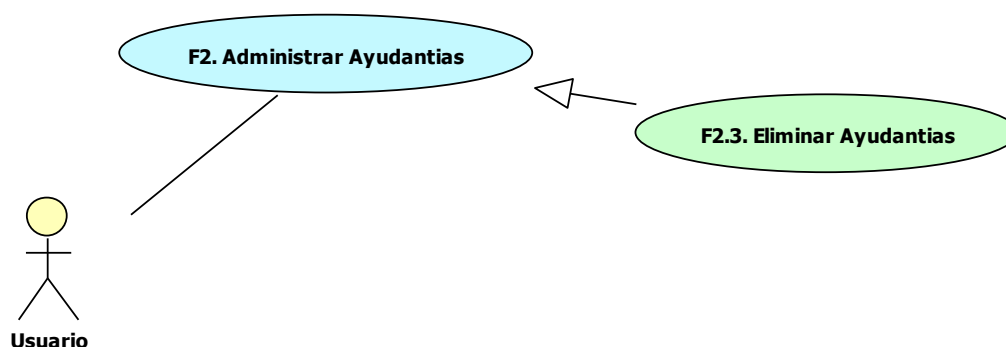
- F2.3. Eliminación de Ayudantías**

*A través de este caso se eliminará una Ayudantía del Sistema.*



## IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

- **Actores:** Usuario



- **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Ayudantías	2	Mostrar ventana de Administración de Ayudantías	
3	Ingresar el identificador de la ayudantía en el TextBox inicial de la ventana.	4	Verificar identificador de la ayudantía en la base de datos	E1
		5	Mostrar datos de la ayudantía en los TextBox restantes de la ventana	
6	Presionar el botón ELIMINAR para borrar la ayudantía	7	Eliminar ayudantía de la base de datos.	E2

### Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

E2	Problemas al Eliminar	Volver a ingresar el ID
----	-----------------------	-------------------------

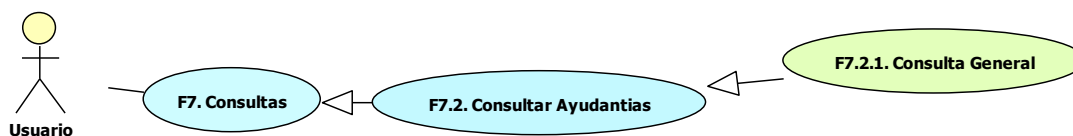
#### Extensiones

Paso	Descripción
4	Si no existe, verificar F2.1

- F7.2.1. Consulta General de Ayudantías**

*A través de este caso de uso se mostrarán todas las ayudantías en el Sistema.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Ayudantías, submenú General	2	Mostrar ventana con tabla de ayudantías	E1

#### Excepciones

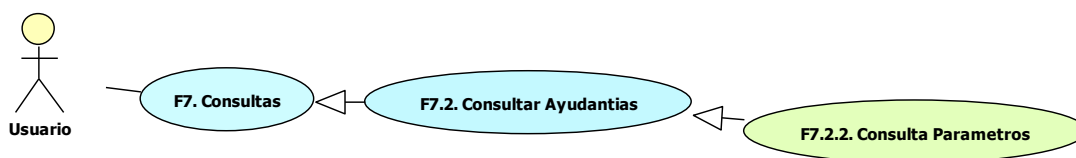
Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- F7.2.2. Consulta por Parámetros de Ayudantías**

*A través de este caso de uso se mostrarán todas las Ayudantías en el Sistema según los parámetros ingresados.*

- Actores:** Usuario

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**



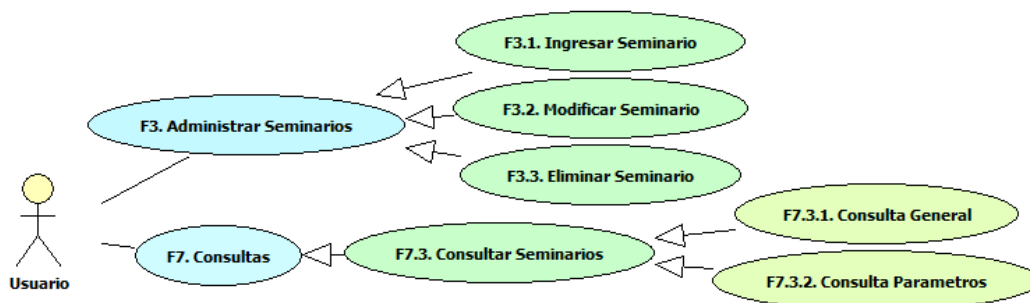
• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Ayudantías, submenú Parámetros	2	Mostrar ventana de Consulta por parámetros de Ayudantías	
3	Seleccionar parámetro en el combobox de la ventana.			
4	Ingresar valor del parámetro			
5	Presionar botón consultar	6	Cargar datos de El ayudantías en la tabla de la ventana	

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

### F3. Administración de seminarios

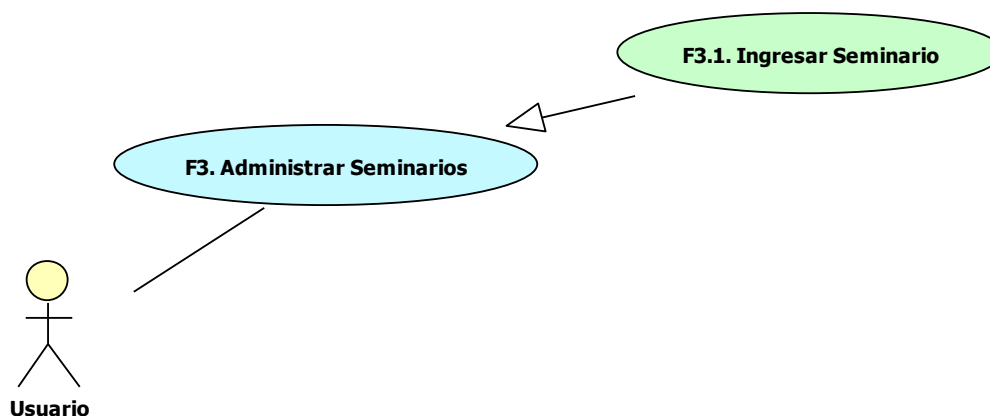


#### Detalle:

- F3.1. Ingreso de Seminarios**

*A través de este caso uso se ingresarán nuevos seminarios de Acción Social*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Seminarios	2	Mostrar ventana de Administración de Seminarios	
3	Ingresar el identificador de seminario en el TextBox inicial de la ventana.	4	Verificar identificador del seminario en la base de datos	E1

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE**

5	Ingresar el resto de datos en la ventana		
6	Presionar el botón GUARDAR para guardar los datos del seminario.	7	Ingresar nuevo E2 Seminario en la base de datos.

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

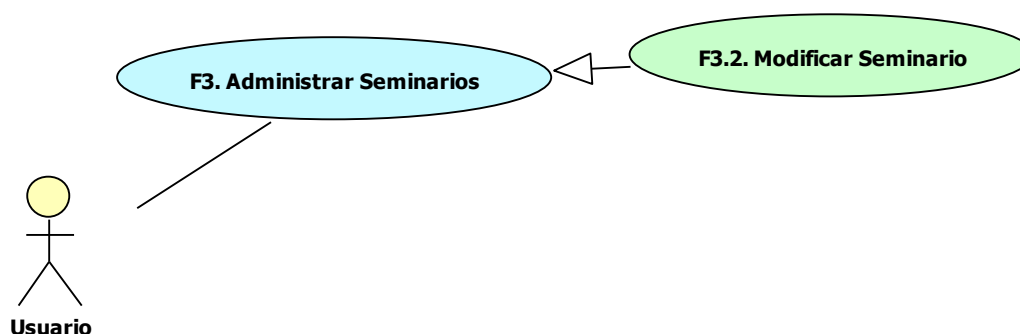
**Extensiones**

Paso	Descripción
4	Si existe, verificar F3.2 o F3.3

- F3.2. Modificación de Seminarios**

*A través de este caso de uso se modificarán todos los datos de un seminario.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú	2	Mostrar ventana de	



**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

3	Administraciones, la opción de Seminarios	Administración de Seminarios
	4	Verificar identificador E1 del seminario en la base de datos
	5	Mostrar datos del seminario en los textbox's restantes de la ventana
6	Modificar datos del seminario en la ventana	
7	Presionar el botón GUARDAR para guardar los datos del seminario	8 Actualizar seminario E2 en la base de datos.

### Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

### Extensiones

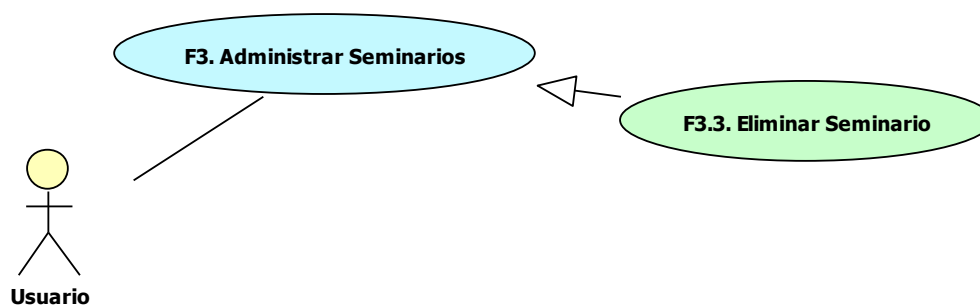
Paso	Descripción
4	Si no existe, verificar F3.1

- F3.3. Eliminación de Seminarios**

*A través de este caso se eliminará un Seminario del Sistema.*

- Actores:** Usuario

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**



• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Seminarios	2	Mostrar ventana de Administración de Seminarios	
3	Ingresar el identificador del seminario en el textbox inicial de la ventana.	4	Verificar identificador del seminario en la base de datos	E1
		5	Mostrar datos del seminario en los textbox restantes de la ventana	
6	Presionar el botón ELIMINAR para borrar el seminario	7	Eliminar seminario de la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Eliminar	Volver a ingresar el ID

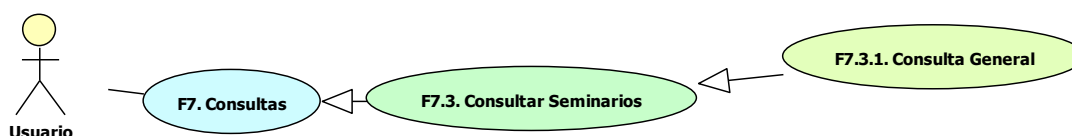
### Extensiones

Paso	Descripción
4	Si no existe, verificar F3.1

- F7.3.1. Consulta General de Seminarios**

*A través de este caso de uso se mostrarán todos los seminarios en el Sistema.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Seminarios, submenú General	2	Mostrar ventana con tabla de Seminarios	E1

### Excepciones

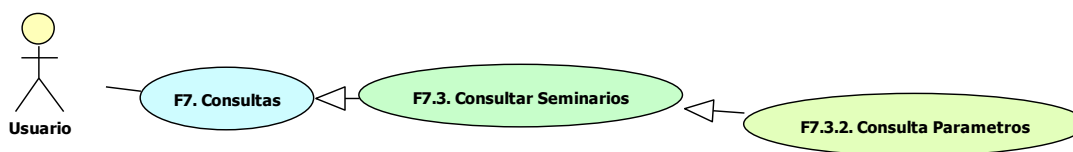
Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- F7.3.2. Consulta por Parámetros de Seminarios**

*A través de este caso de uso se mostrarán todos los Seminarios en el Sistema según los parámetros ingresados.*

- Actores:** Usuario

# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE



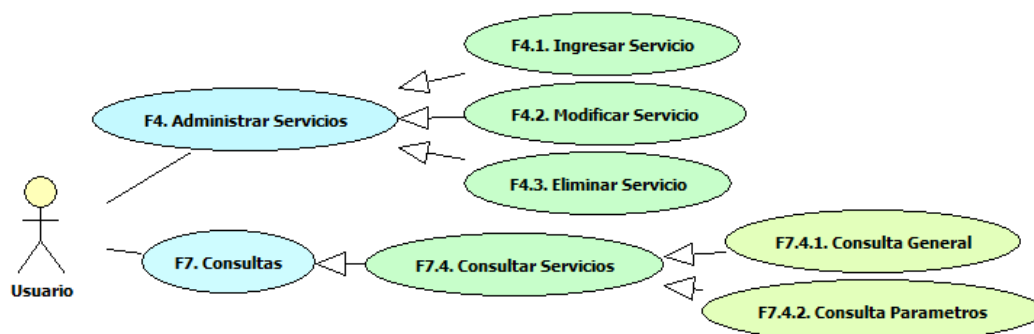
## • Descripción

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Seminarios, submenú Parámetros	2	Mostrar ventana de Consulta parámetros de Seminarios	
3	Seleccionar parámetro en el combobox de la Ventana.			
4	Ingresar valor del parámetro			
5	Presionar botón consultar	6	Cargar datos de El seminarios en la tabla de la ventana	

## Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

#### F4. Administración de servicios

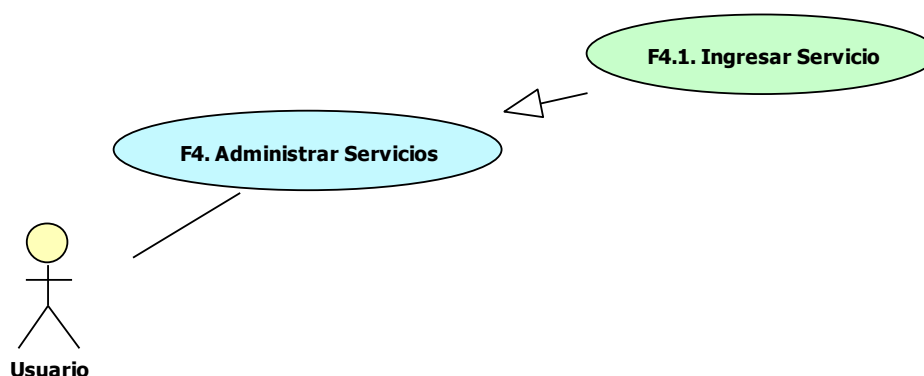


#### Detalle:

- F4.1. Ingreso de Servicios**

*A través de este caso uso se ingresarán nuevos servicios de Acción Social*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Servicios	2	Mostrar ventana de Administración de Servicios	
3	Ingresar el identificador de servicio en el textbox inicial de la ventana.	4	Verificar identificador del servicio en la base de datos	E1

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE**

5	Ingresar el resto de datos en la ventana	
6	Presionar el botón GUARDAR para guardar los datos del servicio.	7 Ingresar nuevo servicio E2 en la base de datos.

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

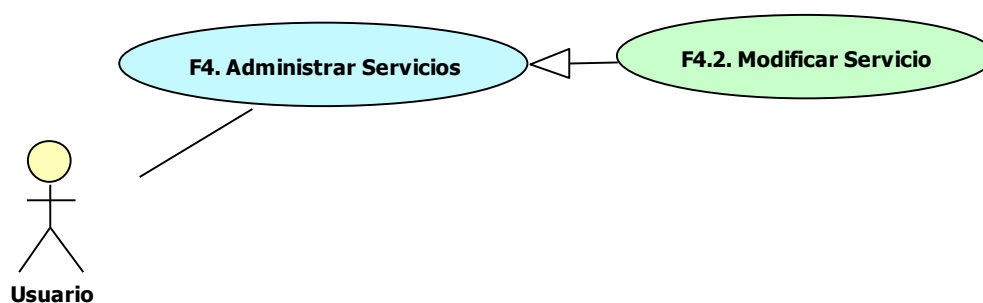
**Extensiones**

Paso	Descripción
4	Si existe, verificar F4.2 o F4.3

• **F4.2. Modificación de Servicios**

*A través de este caso de uso se modificarán todos los datos de un servicio.*

• **Actores:** Usuario



• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú	2	Mostrar ventana de	

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

3	Administraciones, la opción de Servicios	Administración de Servicios
	4	Verificar identificador E1 del servicio en la base de datos
	5	Mostrar datos del servicio en los texttox's restantes de la ventana
6	Modificar datos del servicio en la ventana	
7	Presionar el botón GUARDAR para guardar los datos del servicio	8 Actualizar servicio en E2 la base de datos.

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

**Extensiones**

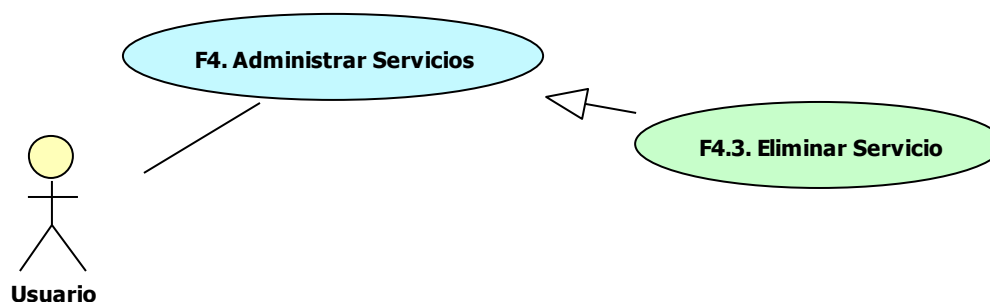
Paso	Descripción
4	Si no existe, verificar F4.1

- F4.3. Eliminación de Servicios**

*A través de este caso se eliminará un Servicio del Sistema.*

- Actores:** Usuario

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**



• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Servicios	2	Mostrar ventana de Administración de Servicios	
3	Ingresar el identificador del servicio en el textbox inicial de la ventana.	4	Verificar identificador del servicio en la base de Datos	E1
		5	Mostrar datos del servicio en los textbox restantes de la ventana	
6	Presionar el botón ELIMINAR para borrar el servicio	7	Eliminar servicio de la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Eliminar	Volver a ingresar el ID

**Extensiones**

Paso	Descripción
4	Si no existe, verificar F4.1

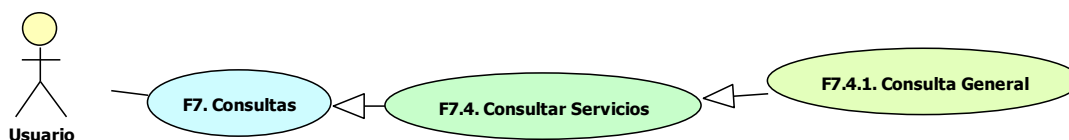




- **F7.4.1. Consulta General de Servicios**

*A través de este caso de uso se mostrarán todos los servicios en el Sistema.*

- **Actores:** Usuario



- **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Servicios, submenú General	2	Mostrar ventana con tabla de servicios	E1

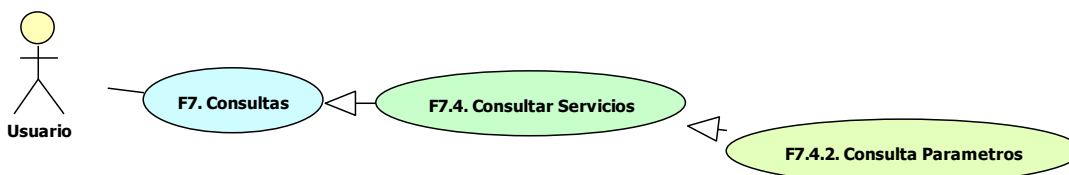
**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- **F7.4.2. Consulta por Parámetros de Servicios**

*A través de este caso de uso se mostrarán todos los Servicios en el Sistema según los parámetros ingresados.*

- **Actores:** Usuario



- **Descripción**

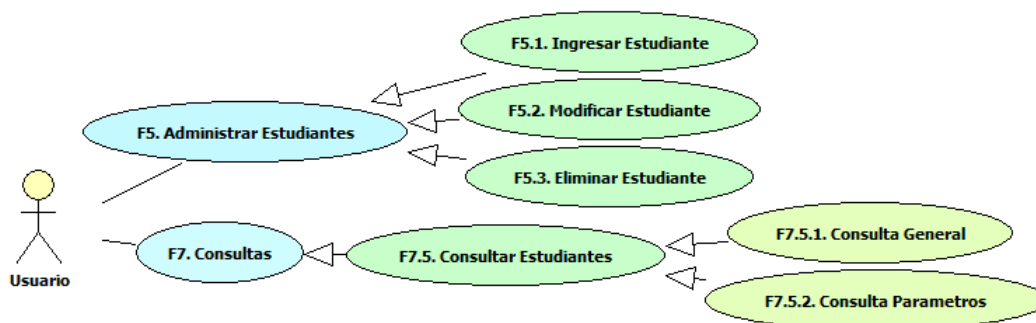
**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Servicios, submenú Parámetros	2	Mostrar ventana de Consulta por Parámetros de Servicios	
3	Seleccionar parámetro en el combobox de la ventana.			
4	Ingresar valor del parámetro			
5	Presionar botón consultar	6	Cargar datos de El servicios en la tabla de la ventana	

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

### F5. Administración de estudiantes

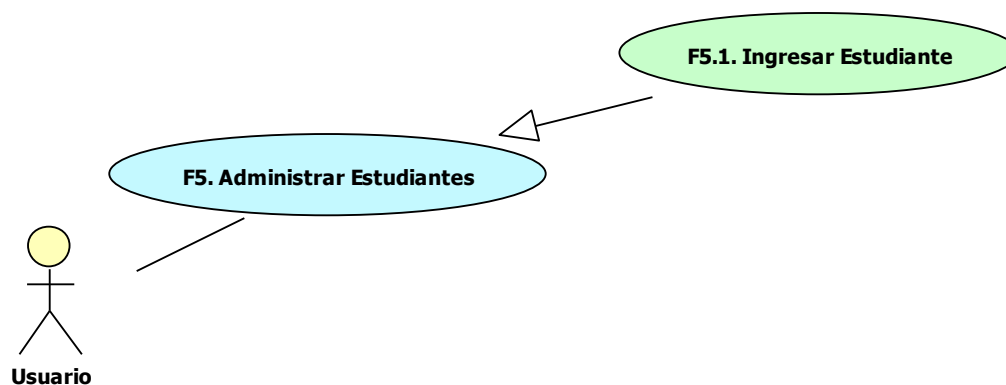


**Detalle:**

- F5.1. Ingreso de Estudiantes**

*A través de este caso uso se ingresarán nuevos estudiantes al Sistema*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Estudiantes	2	Mostrar ventana de Administración de Estudiantes	
3	Ingresar cédula de estudiante en el textbox inicial de la ventana.	4	Verificar identificador del estudiante en la base de datos	E1

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

5	Ingresar el resto de datos en la ventana		
6	Presionar el botón GUARDAR para guardar los datos del estudiante.	7	Ingresar nuevo E2 estudiante en la base de datos.

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

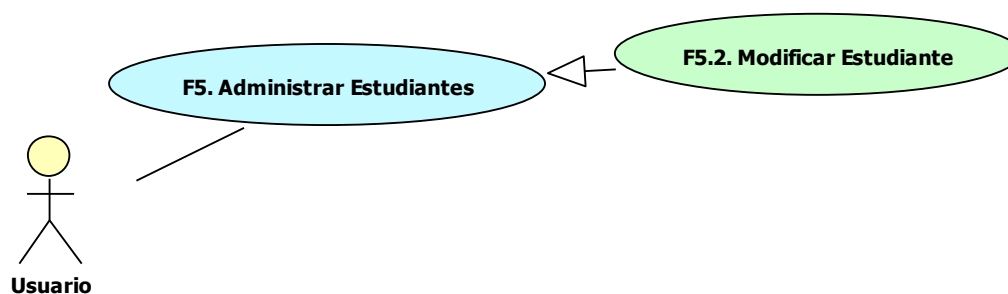
**Extensiones**

Paso	Descripción
4	Si existe, verificar F5.2 o F5.3

- F5.2. Modificación de Estudiantes**

*A través de este caso de uso se modificarán todos los datos de un estudiante.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú	2	Mostrar ventana de	

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

	Administraciones, la opción de Estudiantes	Administración de Estudiantes
3	Ingresar cédula del estudiante en el textbox inicial de la ventana.	4 Verificar identificador E1 del estudiante en la base de datos
		5 Mostrar datos del estudiante en los textbox's restantes de la ventana
6	Modificar datos del estudiante en la ventana	
7	Presionar el botón GUARDAR para guardar los datos del estudiante	8 Actualizar estudiante E2 en la base de datos.

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

**Extensiones**

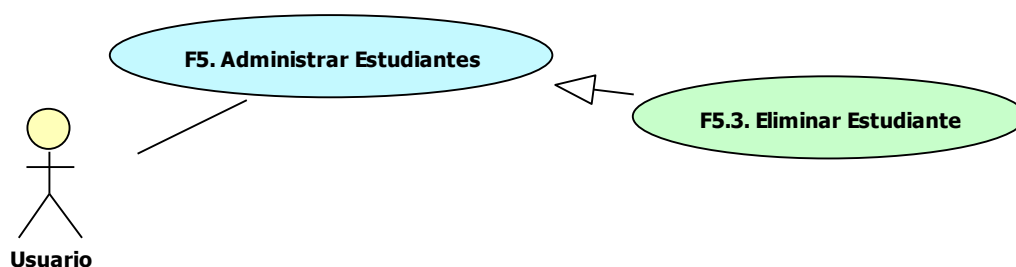
Paso	Descripción
4	Si no existe, verificar F5.1

- F5.3. Eliminación de Estudiantes**

*A través de este caso se eliminará un estudiante del Sistema.*

- Actores:** Usuario

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**



• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Estudiantes	2	Mostrar ventana de Administración de Estudiantes	
3	Ingresar cédula del estudiante en el textbox inicial de la ventana.	4	Verificar identificador del estudiante en la base de datos	E1
		5	Mostrar datos del estudiante en los textbox restantes de la ventana	
6	Presionar el botón ELIMINAR para borrar el estudiante	7	Eliminar estudiante de la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Eliminar	Volver a ingresar cédula

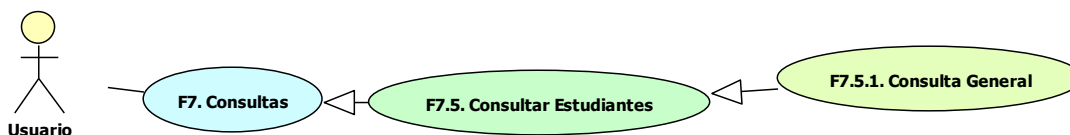
## Extensiones

Paso	Descripción
4	Si no existe, verificar F5.1

- F7.5.1. Consulta General de Estudiantes**

*A través de este caso de uso se mostrarán todos los estudiantes en el Sistema.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Estudiantes, submenú General	2	Mostrar ventana con tabla de estudiantes	E1

## Excepciones

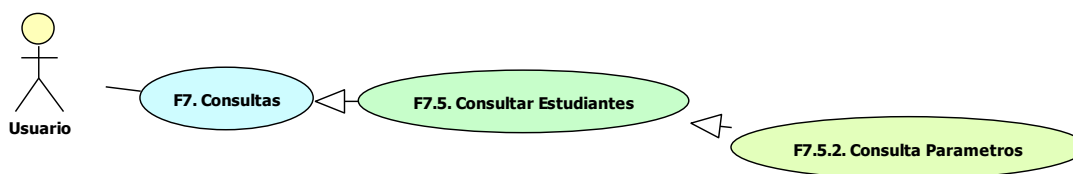
Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- F7.5.2. Consulta por Parámetros de Estudiantes**

*A través de este caso de uso se mostrarán todos los estudiantes en el Sistema según los parámetros ingresados.*

- Actores:** Usuario

# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE



## • Descripción

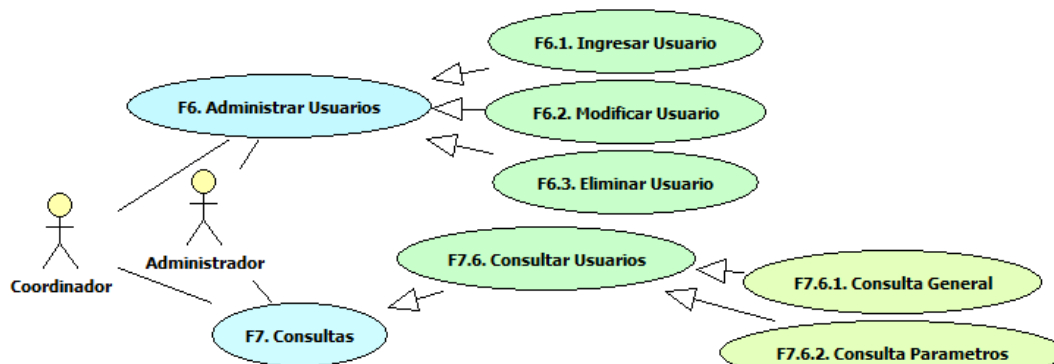
Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Estudiantes, submenú Parametros	2	Mostrar ventana de Consulta por Parametros de Estudiantes	
3	Seleccionar parámetro en el combobox de la ventana.			
4	Ingresar valor del parámetro			
5	Presionar botón consultar	6	Cargar datos de El estudiantes en la tabla de la ventana	

## Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión



### F6. Administración de usuarios

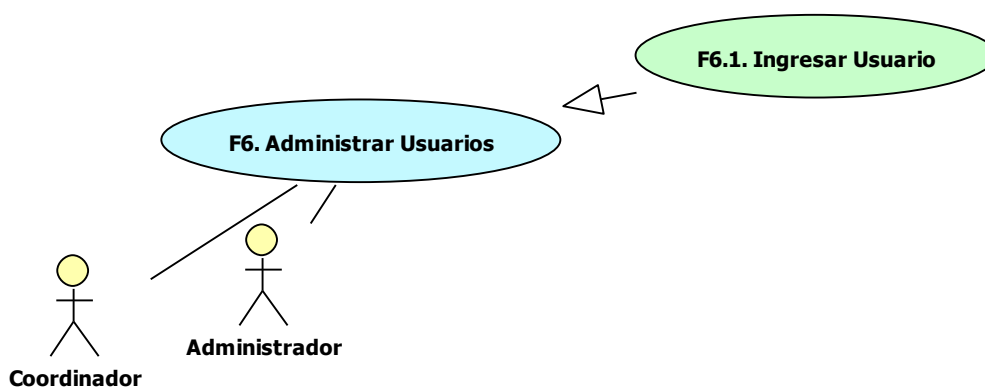


#### Detalle:

- F6.1. Ingreso de Usuarios**

A través de este caso uso se ingresarán nuevos usuarios al Sistema

- Actores:** Coordinador, Administrador



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú de Administraciones, la opción de Usuarios	2	Mostrar ventana de Administración de Usuarios	
3	Ingresar el identificador de usuario en el textbox inicial	4	Verificar identificador del usuario en la base	E1

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

	de la ventana.	de datos
<b>5</b>	Ingresar el resto de datos en la ventana	
<b>6</b>	Presionar el botón GUARDAR para guardar los datos del usuario.	<b>7</b> Ingresar nuevo usuario E2 en la base de datos.

**Excepciones**

Código	Descripción	Alternativa
<b>E1</b>	No hay conexión a BDD	Reintentar conexión
<b>E2</b>	Problemas al Guardar	Pedir reingreso de datos

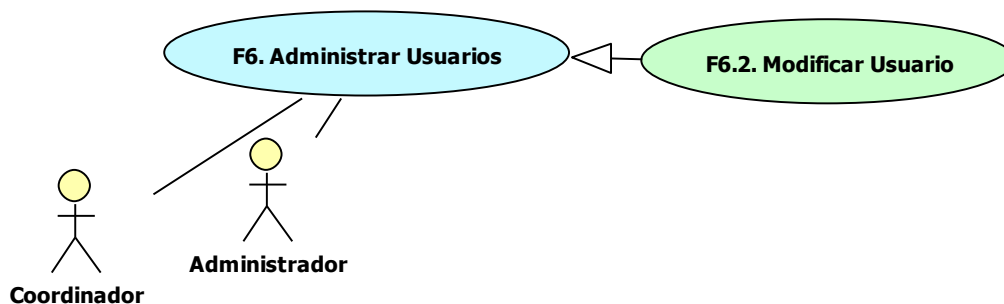
**Extensiones**

Paso	Descripción
<b>4</b>	Si existe, verificar F6.2 o F6.3

- F6.2. Modificación de Usuarios**

*A través de este caso de uso se modificarán todos los datos de un usuario.*

- Actores:** Coordinador, Administrador



**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Usuarios	2	Mostrar ventana de Administración de Usuarios	
3	Ingresar el identificador del usuario en el textbox inicial de la ventana.	4	Verificar identificador del usuario en la base de datos	E1
		5	Mostrar datos del usuario en los textbox's restantes de la ventana	
6	Modificar datos del usuario en la ventana			
7	Presionar el botón GUARDAR para guardar los datos del usuario	8	Actualizar usuario en la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Guardar	Pedir reingreso de datos

**Extensiones**

Paso	Descripción
4	Si no existe, verificar F6.1

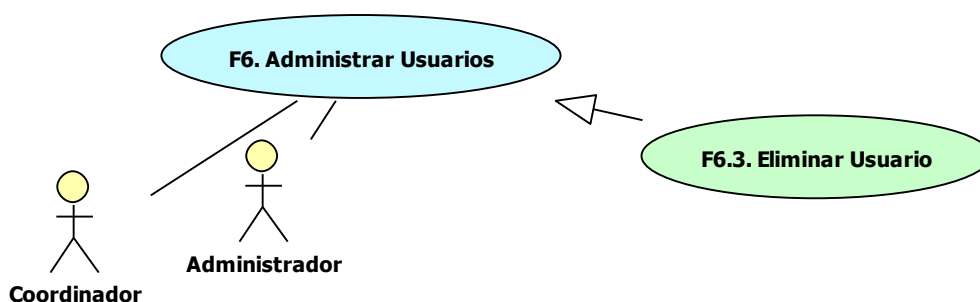
- F6.3. Eliminación de Usuarios**

*A través de este caso se eliminará un usuario del Sistema.*



**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

- **Actores:** Coordinador, Administrador



- **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Administraciones, la opción de Usuarios	2	Mostrar ventana de Administración de Usuarios	
3	Ingresar el identificador del usuario en el textbox inicial de la ventana.	4	Verificar identificador del usuario en la base de datos	E1
		5	Mostrar datos del usuario en los textbox restantes de la ventana	
6	Presionar el botón ELIMINAR para borrar el usuario	7	Eliminar usuario de la base de datos.	E2

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Problemas al Eliminar	Volver a ingresar el ID

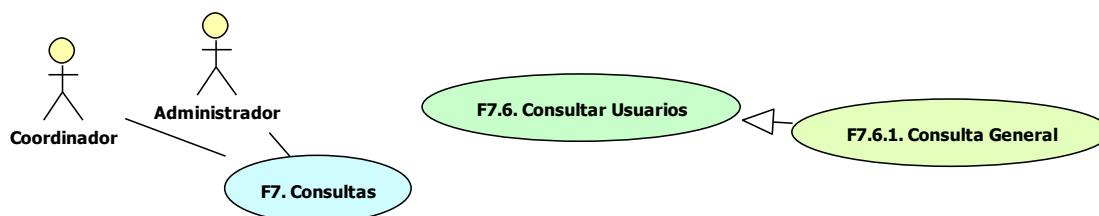
## Extensiones

Paso	Descripción
4	Si no existe, verificar F6.1

- F7.6.1. Consulta General de Usuarios**

*A través de este caso de uso se mostrarán todos los usuarios en el Sistema.*

- Actores:** Coordinador, Administrador



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Usuarios, submenú General	2	Mostrar ventana con tabla de usuarios	E1

## Excepciones

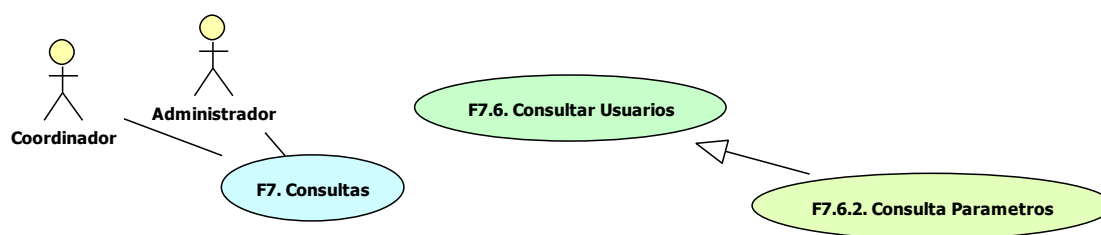
Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- F7.6.2. Consulta por Parámetros de Usuarios**

*A través de este caso de uso se mostrarán todos los usuarios en el Sistema según los parámetros ingresados.*

- Actores:** Coordinador, Administrador

# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE



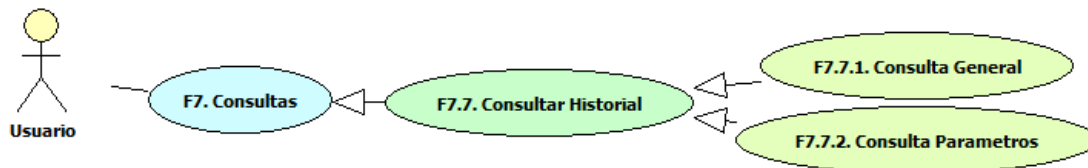
## • Descripción

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Usuarios, submenú Parámetros	2	Mostrar ventana de Consulta por Parámetros de Usuarios	
3	Seleccionar parámetro en el combobox de la ventana.			
4	Ingresar valor del parámetro			
5	Presionar botón consultar	6	Cargar datos de El usuarios en la tabla de la ventana	

## Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

### *F7.7. Consulta de historial*

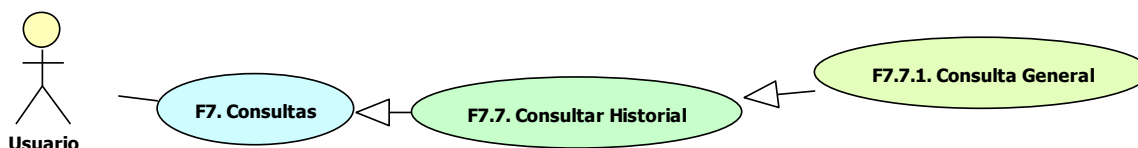


#### **Detalle:**

- F7.7.1. Consulta General del Historial**

*A través de este caso de uso se mostrarán todos sucesos registrados en el historial de eventos del Sistema*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Historial, submenú General	2	Mostrar ventana con tabla del historial	E1

#### **Excepciones**

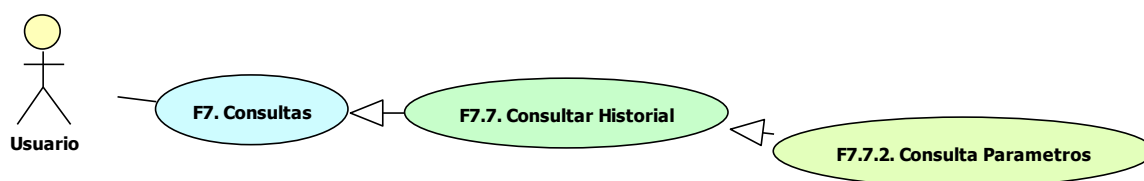
Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- F7.6.2. Consulta por Parámetros de Historial**

*A través de este caso de uso se mostrarán todos sucesos registrados en el historial de eventos del Sistema según los parámetros ingresados.*

- Actores:** Usuario

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**



• **Descripción**

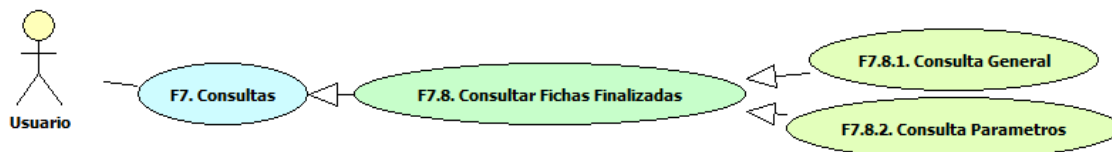
Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Historial, submenú Parámetros	2	Mostrar ventana de Consulta Parámetros del Historial	
3	Seleccionar parámetro en el combobox de la ventana.			
4	Ingresar valor del parámetro			
5	Presionar botón consultar	6	Cargar datos del historial en la tabla de la ventana	E1

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión



### F7.8. Consulta de fichas finalizadas

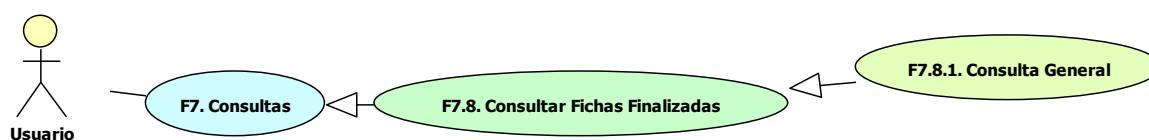


#### Detalle:

- F7.8.1. Consulta General de Fichas Finalizadas**

A través de este caso de uso se mostrarán todas las fichas de acción social estudiantil con estado finalizadas.

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Fichas Finalizadas, submenú General	2	Mostrar ventana con tabla de las Fichas Finalizadas	E1

#### Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

- F7.8.2. Consulta por Parámetros de Historial**

A través de este caso de uso se mostrarán todas las fichas de acción social estudiantil con estado finalizadas según los parámetros ingresados.

- Actores:** Usuario

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**



• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Consultas, la opción Fichas Finalizadas, submenú Parámetros	2	Mostrar ventana de Consulta por Parámetros de las Fichas Finalizadas	
3	Seleccionar parámetro en el combobox de la ventana.			
4	Ingresar valor del parámetro			
5	Presionar botón consultar	6	Cargar datos de las fichas finalizadas en la tabla de la ventana	E1

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

### *F8. Gestión de fichas de acción social*



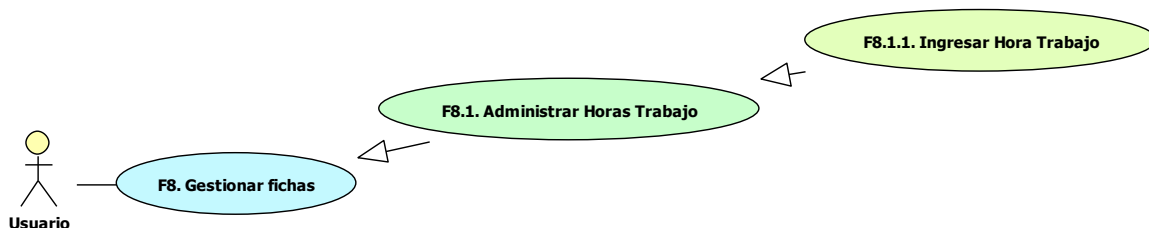
#### **Detalle:**

#### **F8.1. Administración de horas de trabajos**

- F8.1.1. Ingreso de Horas**

*A través de este caso uso se ingresarán nuevas horas de Acción Social a la ficha del estudiante*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú	2	Mostrar ventana de	
	Procesos, la opción de		Gestión de la Ficha del	
	Ficha de Horas		Estudiante	
3	Ingresar la identificación	4	Verificar trabajos del E1	
	del estudiante en el textbox		estudiante en la base de	
	para cédula		datos	
5	Seleccionar tipo de trabajo	6	Cargar trabajos E2	
			disponibles por tipo	

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

7	Seleccionar el trabajo, fechas de asignación, horas y observaciones	
8	Seleccionar submenú Trabajo>Agregar o presionar F12	9 Ingresar registro de E3, E4 horas de Acción Social del estudiante

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	No existen trabajos	Ingresar trabajos (F2 – F5)
E3	Problemas al Guardar	Pedir reingreso de datos
E4	Ficha Finalizada	No se puede ingresar más registros de horas

**Extensiones**

Paso	Descripción
4	Si existe trabajos, también verificar F8.1.2 o F8.1.3
9	Si la ficha está finalizada, no se puede ingresar el registro de horas

- F8.1.2. Modificación de Horas**

*A través de este caso uso se modificará el detalle de las horas de Acción Social en la ficha del estudiante.*

- Actores:** Usuario



**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Procesos, la opción de Ficha de Horas	2	Mostrar ventana de Gestión de la Ficha del Estudiante	
3	Ingresar la identificación del estudiante en el textbox para cédula	4	Verificar trabajos del estudiante en la base de datos	E1
5	Seleccionar el registro del trabajo a modificar			
6	Click derecho>Editar	7	Cargar datos del trabajo existente	E2
8	Modificar el trabajo, fechas de asignación, horas u observaciones			
9	Seleccionar submenú Trabajo>Modificar o presionar F12	10	Modificar registro de horas de Acción Social del estudiante	E3, E4

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Trabajo ya no existe	Ingresar trabajo (F2 – F5)
E3	Problemas al Guardar	Pedir reingreso de datos
E4	Ficha Finalizada	No se puede modificar el registro de horas

### Extensiones

Paso	Descripción
4	Si no existen trabajos, verificar F8.1.1
9	Si la ficha está finalizada, no se puede modificar el registro de horas

- F8.1.3. Eliminación de Horas**

*A través de este caso uso se eliminarán detalles de las horas de Acción Social en la ficha del estudiante.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Procesos, la opción de Ficha de Horas	2	Mostrar ventana de Gestión de la Ficha del Estudiante	
3	Ingresar la identificación del estudiante en el TextBox para cédula	4	Verificar trabajos del estudiante en la base de datos	E1
5	Seleccionar el registro del trabajo a eliminar			
6	Click derecho>Eliminar	7	Cargar datos del trabajo existente	E2
8	Presionar el botón SI en el dialogo de confirmación	9	Eliminar registro de horas de Acción Social	E3, E4

	del estudiante
--	----------------

### Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Trabajo ya no existe	Ingresar trabajo (F2 – F5)
E3	Problemas al Guardar	Pedir reingreso de datos
E4	Ficha Finalizada	No se puede eliminar el registro de horas

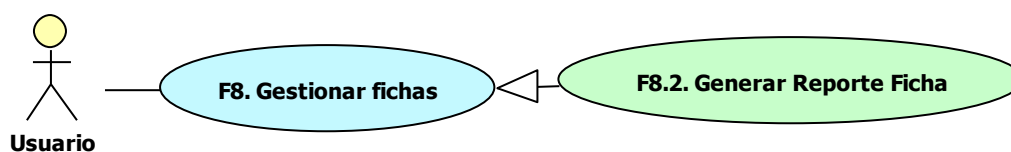
### Extensiones

Paso	Descripción
4	Si no existen registros, verificar F8.1.1
9	Si la ficha está finalizada, no se puede eliminar el registro de horas

### F8.2. Generar reporte de ficha

A través de este caso de uso se generará y visualizará el reporte de la ficha de Acción Social Universitaria del estudiante.

- **Actores:** Usuario



- **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Procesos, la opción de Ficha de Horas	2	Mostrar ventana de Gestión de la Ficha del Estudiante	

3	Ingresar la identificación del estudiante en el textbox para cédula	4	Verificar trabajos del E1 estudiante en la base de datos
5	Seleccionar submenú Reporte>Generar o presionar F9		
6	Seleccionar ficha para estudiantes antiguos o nuevos (AGAI-ASUI)	7	Visualizar el reporte de la ficha de Acción Social del estudiante

### Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión

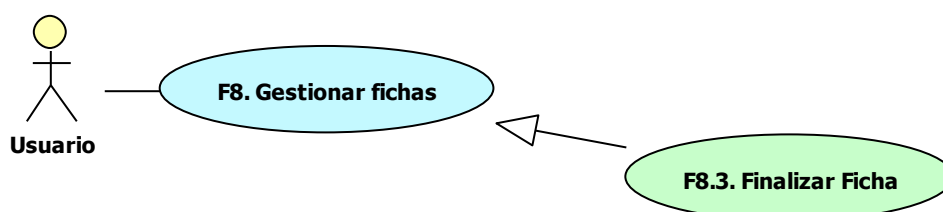
### Extensiones

Paso	Descripción
7	Si no existen registros de horas, verificar F8.1.1

### F8.3. Finalizar ficha

*A través de este caso de uso se finalizará (cerrará) la ficha de Acción Social Universitaria del estudiante.*

- **Actores:** Usuario





**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

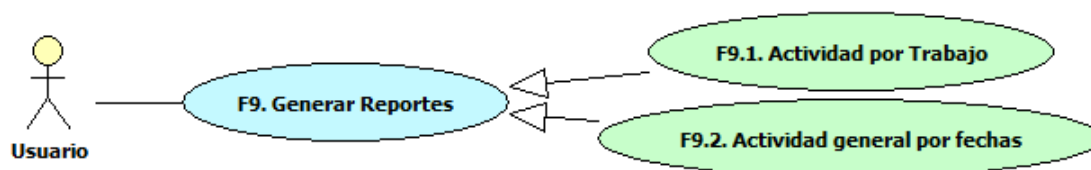
• **Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Procesos, la opción de Ficha de Horas	2	Mostrar ventana de Gestión de la Ficha del Estudiante	
3	Ingresar la identificación del estudiante en el textbox para cédula	4	Verificar trabajos del estudiante en la base de datos	E1
5	Seleccionar submenú Estudiante>Finalizar o presionar F5			E2
6	Aceptar la ventana de confirmación	7	Finalizar la ficha de Acción Social del estudiante	

**Excepciones**

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	Ficha Finalizada	La ficha ya está Finalizada

### *F9. Generación de reportes de actividades*

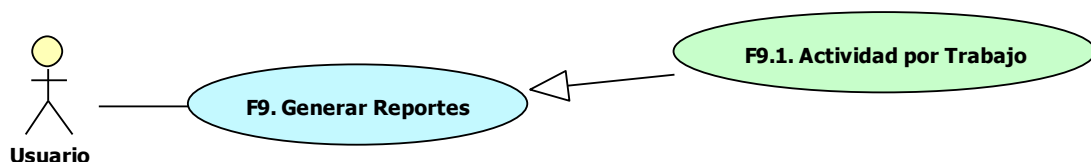


#### **Detalle:**

- F9.1. Generar reportes de actividad por trabajo**

*A través de este caso uso se generarán y visualizarán reportes de actividades por trabajos asignados.*

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Realizar una consulta general o por parámetros de un trabajo (Proyectos, Ayudantías, Seminarios, Servicios)	2	Mostrar ventana de resultados de la consulta	E1
3	Seleccionar un registro			E2
4	Click derecho>Estudiantes Asignados	5	Generar y mostrar ventana con el reporte solicitado	

## Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	No existen trabajos	Ingresar trabajos (F2 – F5)

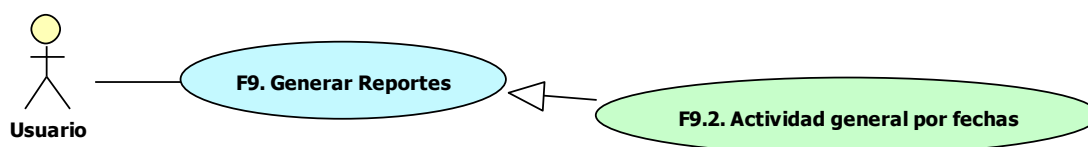
## Extensiones

Paso	Descripción
2	Si no existen registros, verificar ingreso F2 – F5

- F9.2. Generar reportes de actividad por fechas**

A través de este caso uso se generará y visualizará un reporte de actividades generales de todos los trabajos entre un rango de fechas.

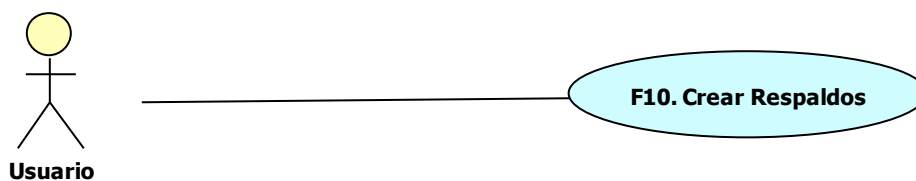
- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Procesos, la opción Actividad de Trabajos	2	Mostrar ventana de ingreso de rango de fechas	
3	Ingresar el rango de fechas a filtrar los resultados	4	Generar y mostrar ventana con el reporte solicitado	

### F10. Creación de respaldos de datos

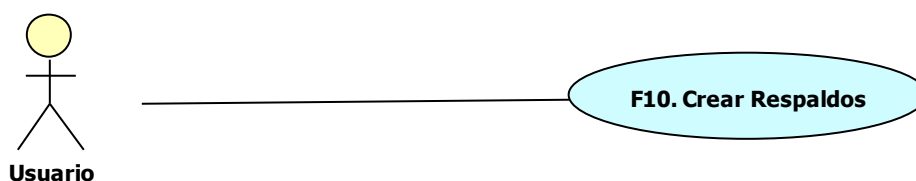


#### Detalle:

- F10. Crear respaldos de datos**

A través de este caso uso se creará un archivo comprimido con los respaldos necesarios de la información manejada por el Sistema.

- Actores:** Usuario



- Descripción**

Paso	Actor	Paso	Sistema	Excepción
1	Seleccionar del menú Sistema, submenú Respaldo de Datos	2	Mostrar ventana de selección del destino del archivo de respaldo	
3	Seleccionar destino y presionar en Guardar	4	Generar archivo comprimido con el respaldo solicitado	E1, E2

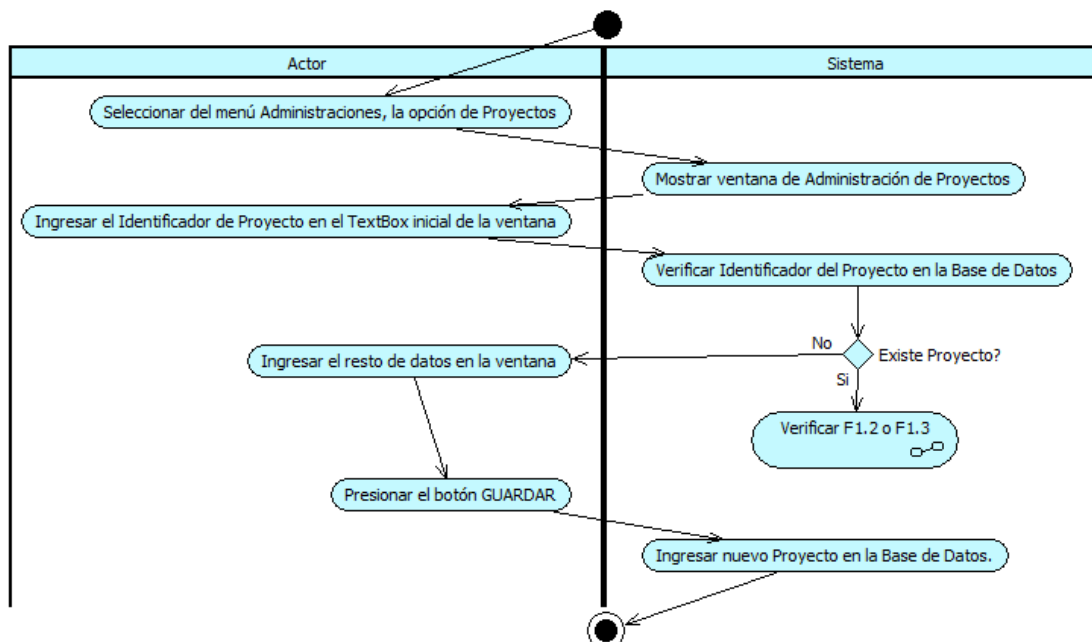
#### Excepciones

Código	Descripción	Alternativa
E1	No hay conexión a BDD	Reintentar conexión
E2	No existen permisos de escritura	Intentar otra ubicación

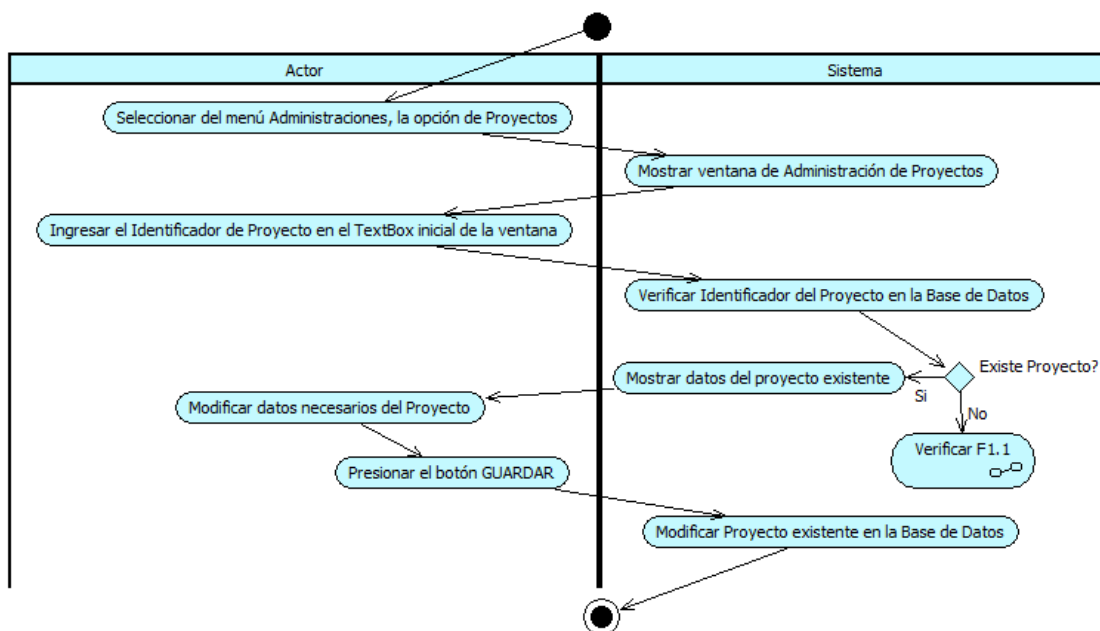
### 5.1.3 Diagramas de Actividades

#### Manejo de Proyectos

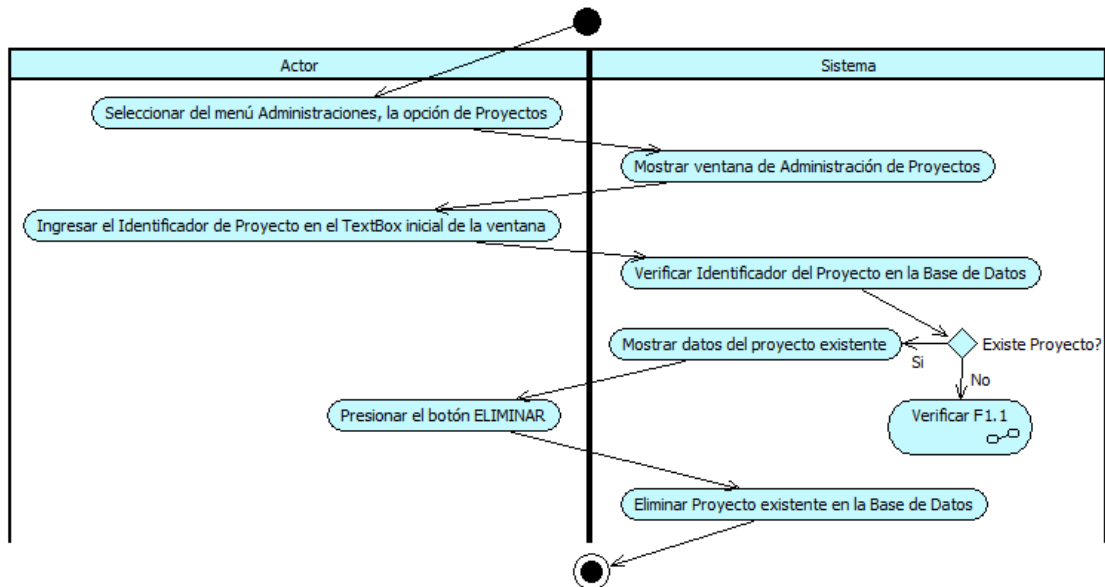
##### F1.1. Ingreso de Proyectos



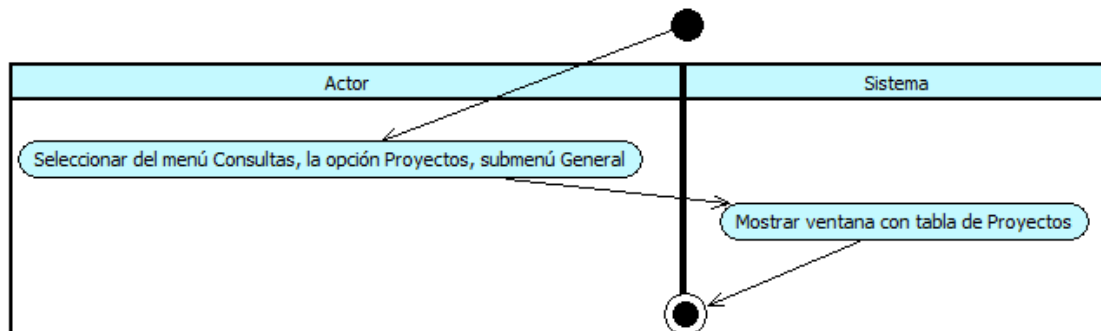
##### F1.2. Modificación de Proyectos



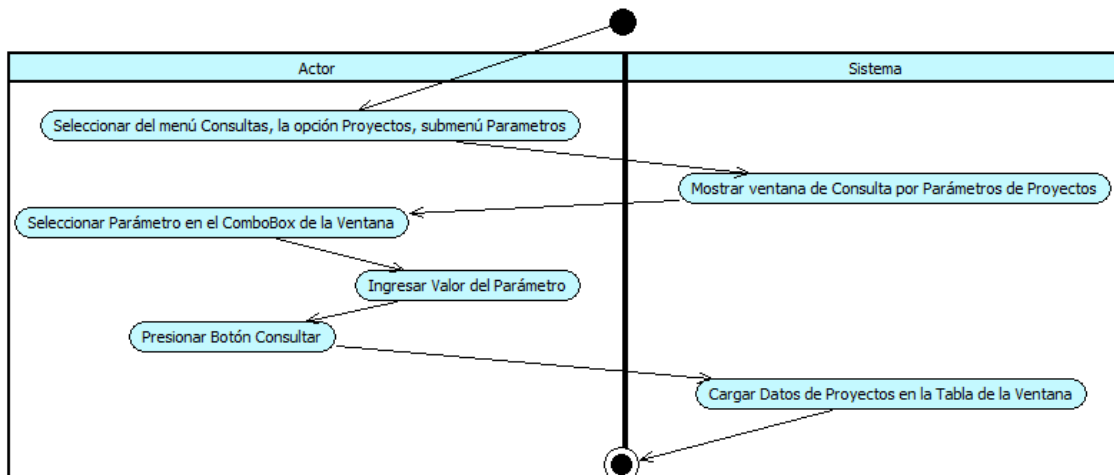
### F1.3. Eliminación de Proyectos



### F7.1.1. Consulta general de Proyectos

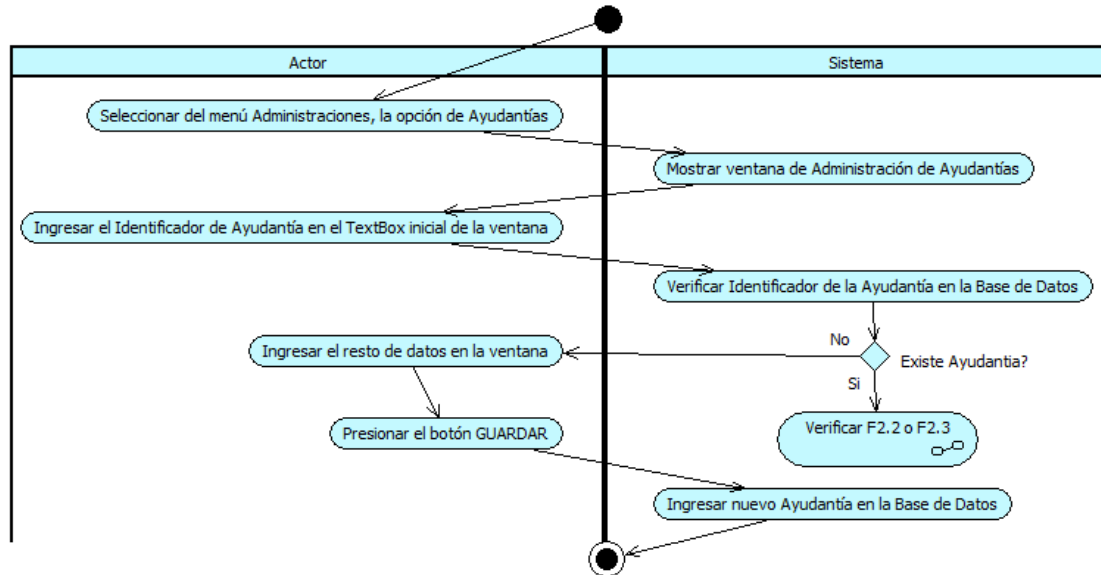


### F7.1.2. Consulta por Parámetros de Proyectos

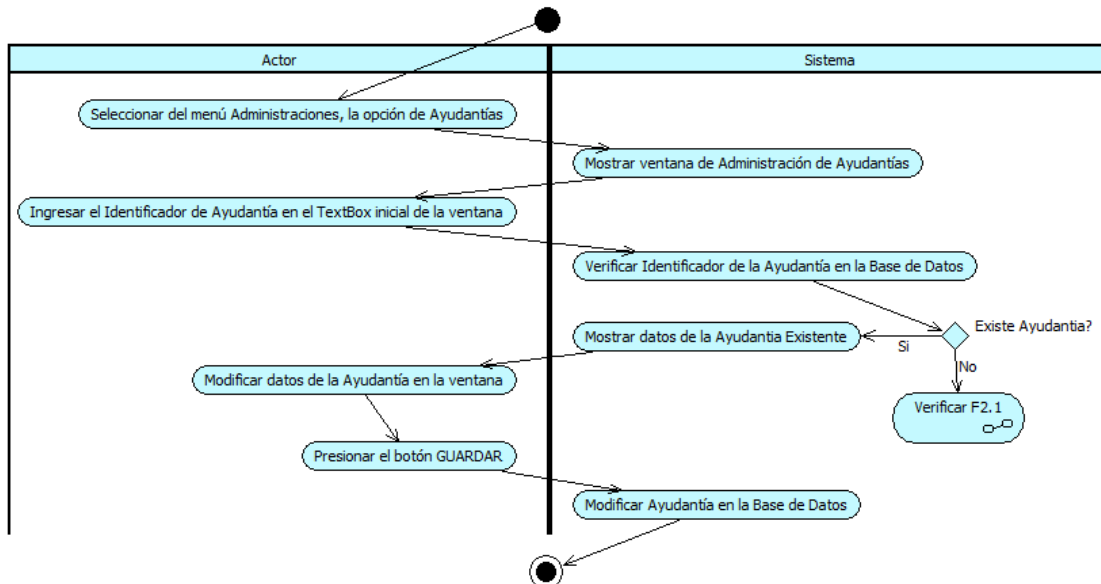


## Manejo de Ayudantías

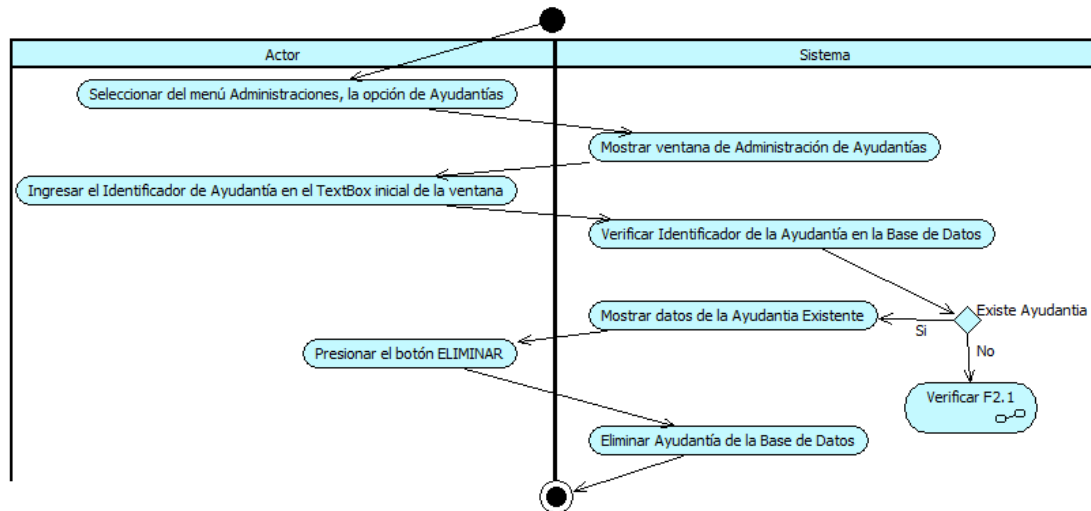
### F2.1. Ingreso de Ayudantías



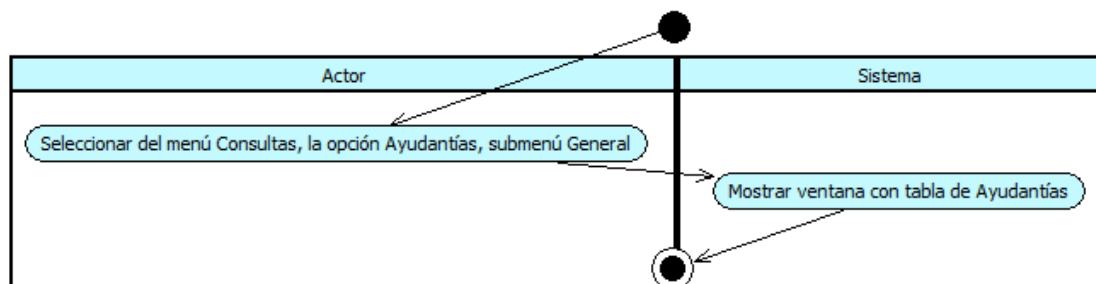
### F2.2. Modificación de Ayudantías



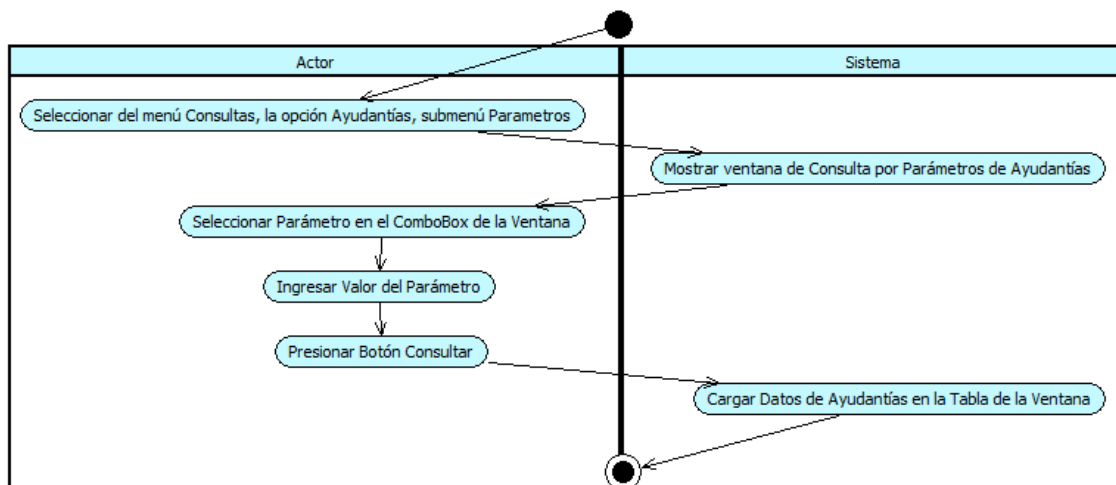
### F2.3. Eliminación de Ayudantías



### F7.2.1. Consulta general de Ayudantías



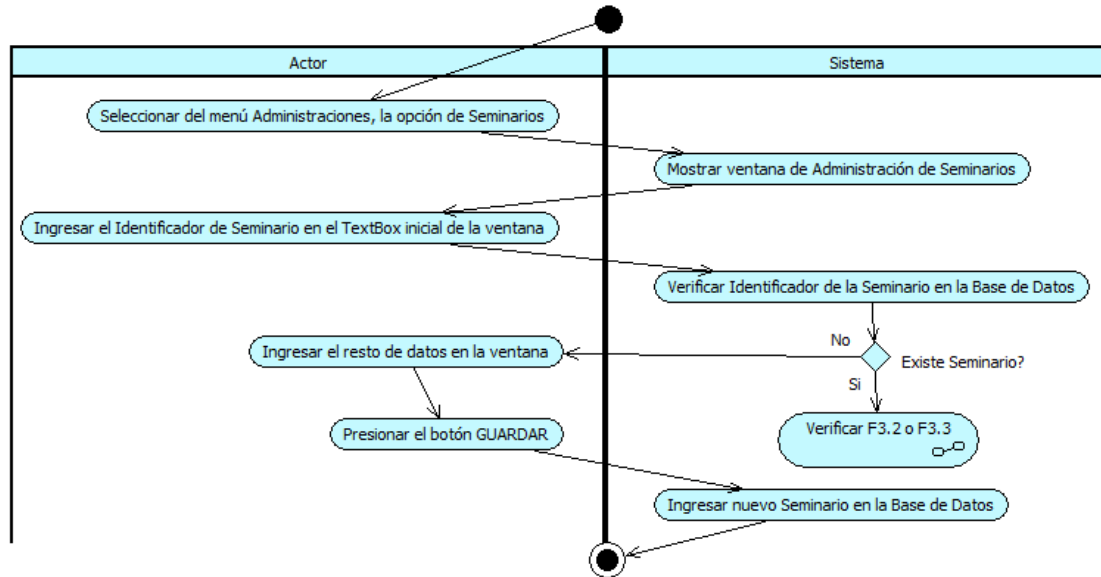
### F7.2.2. Consulta por Parámetros de Ayudantías



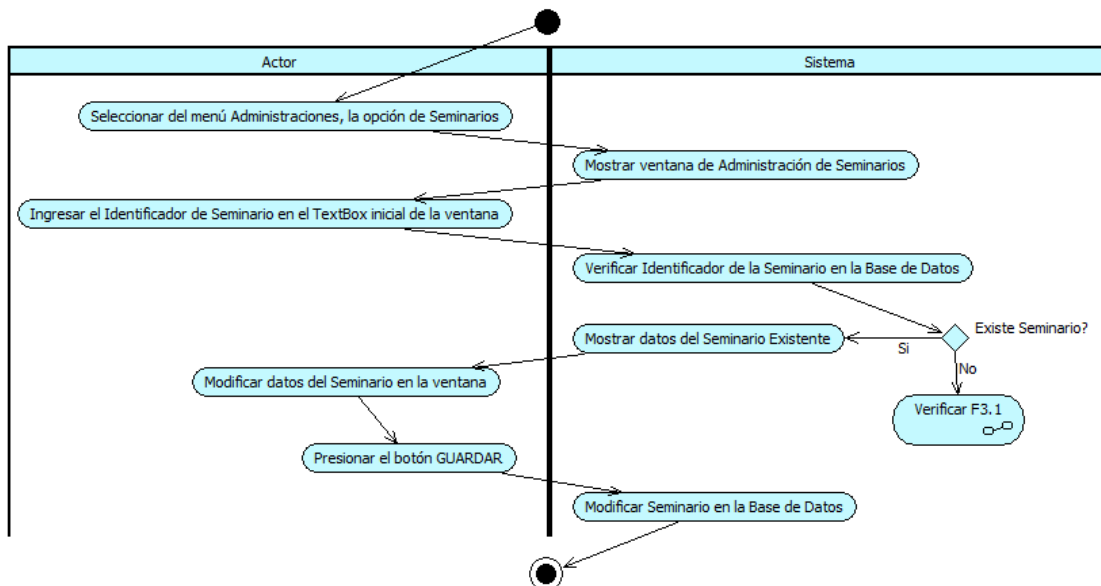


## Manejo de Seminarios

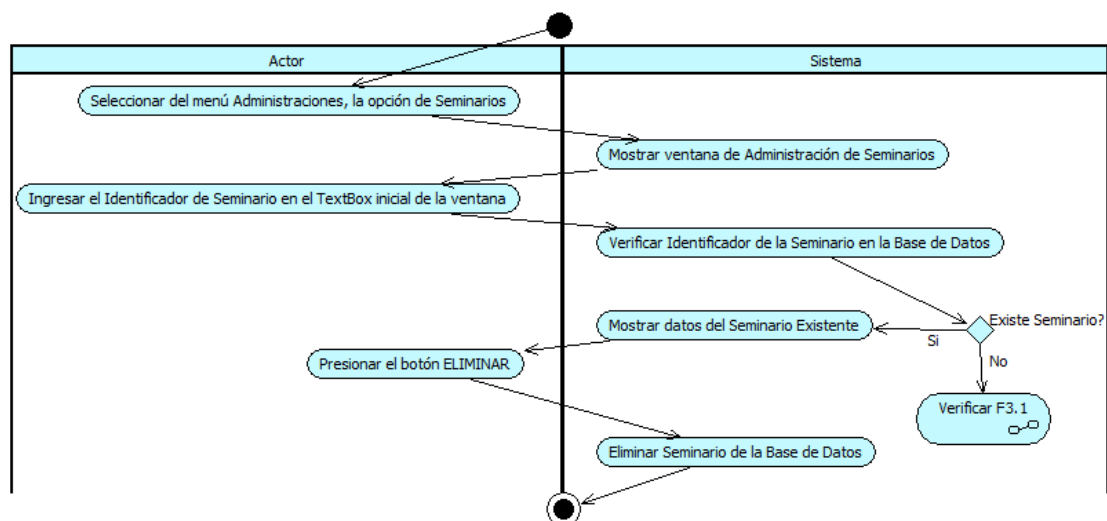
### F3.1. Ingreso de Seminarios



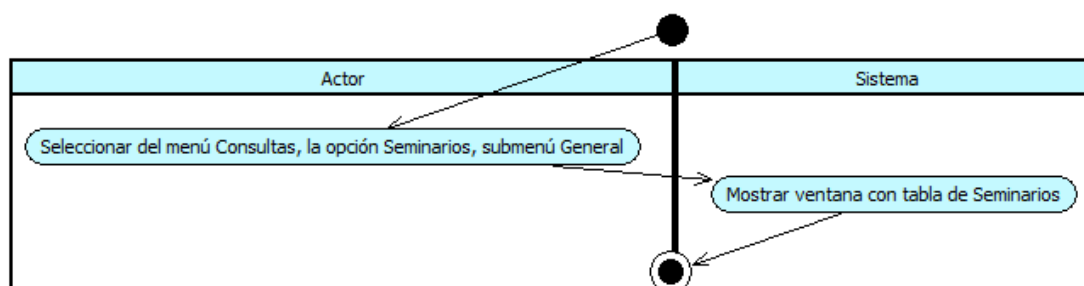
### F3.2. Modificación de Seminarios



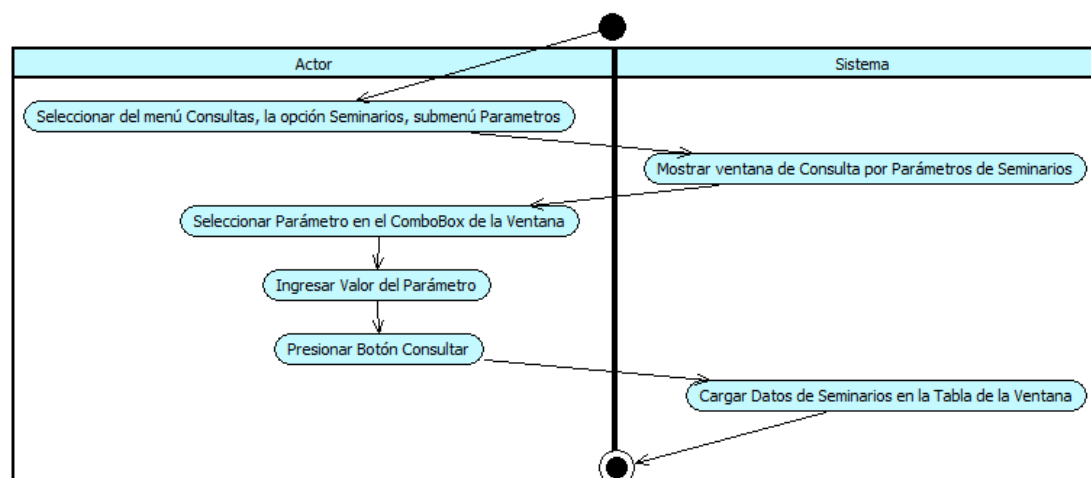
### F3.3. Eliminación de Seminarios



#### F7.3.1. Consulta general de Seminarios

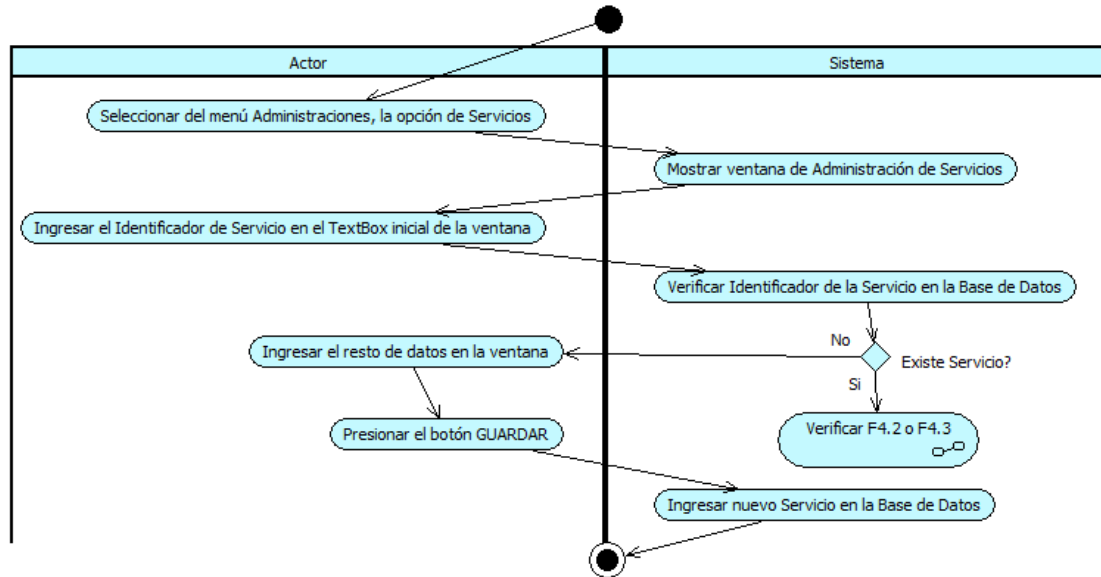


#### F7.3.2. Consulta por Parámetros de Seminarios

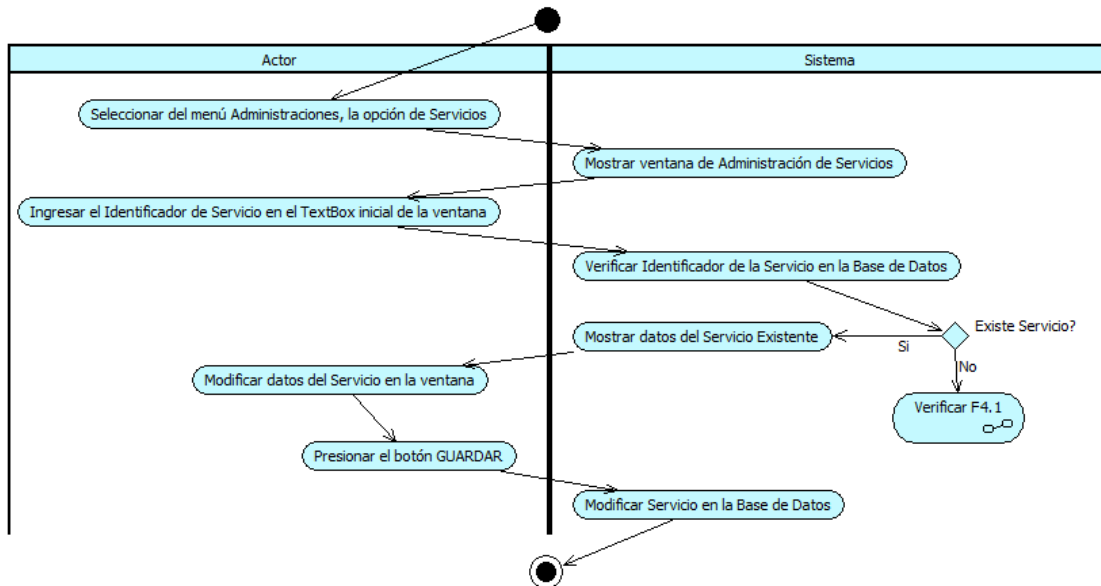


### Manejo de Servicios

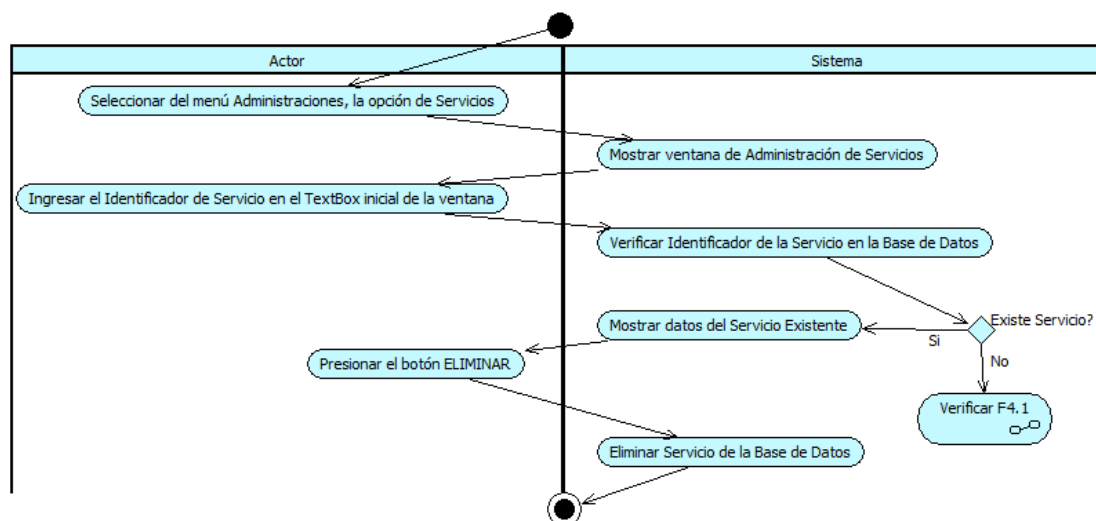
#### F4.1. Ingreso de Servicios



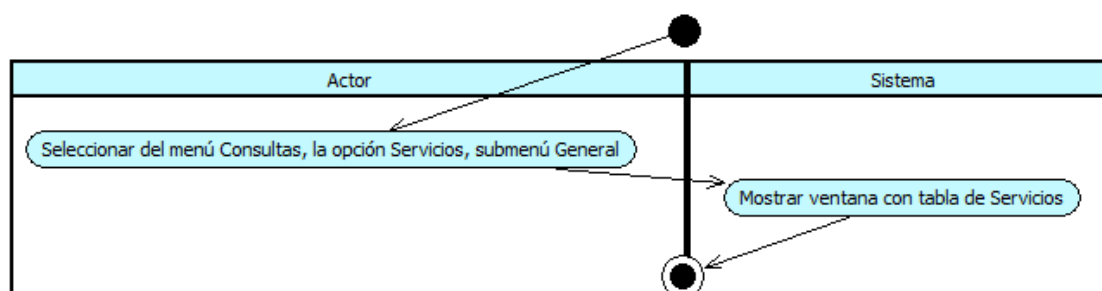
#### F4.2. Modificación de Servicios



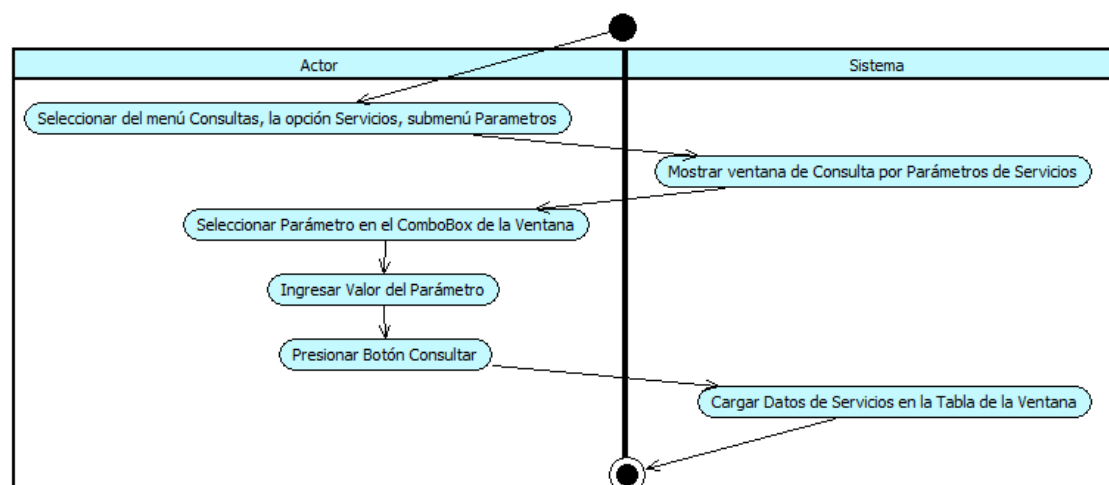
### F4.3. Eliminación de Servicios



### F7.4.1. Consulta general de Servicios

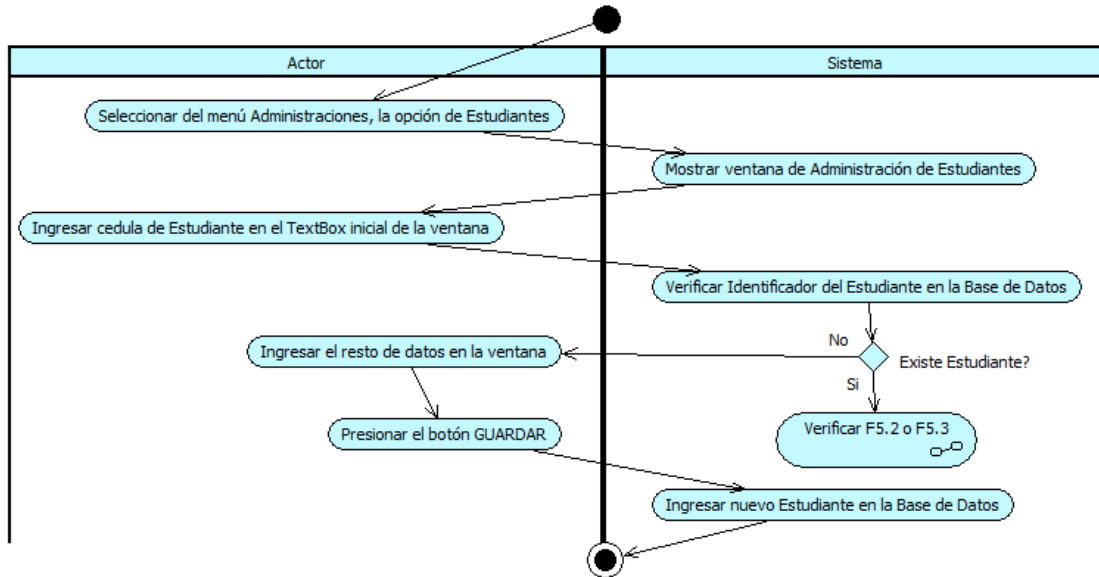


### F7.4.2. Consulta por Parámetros de Servicios

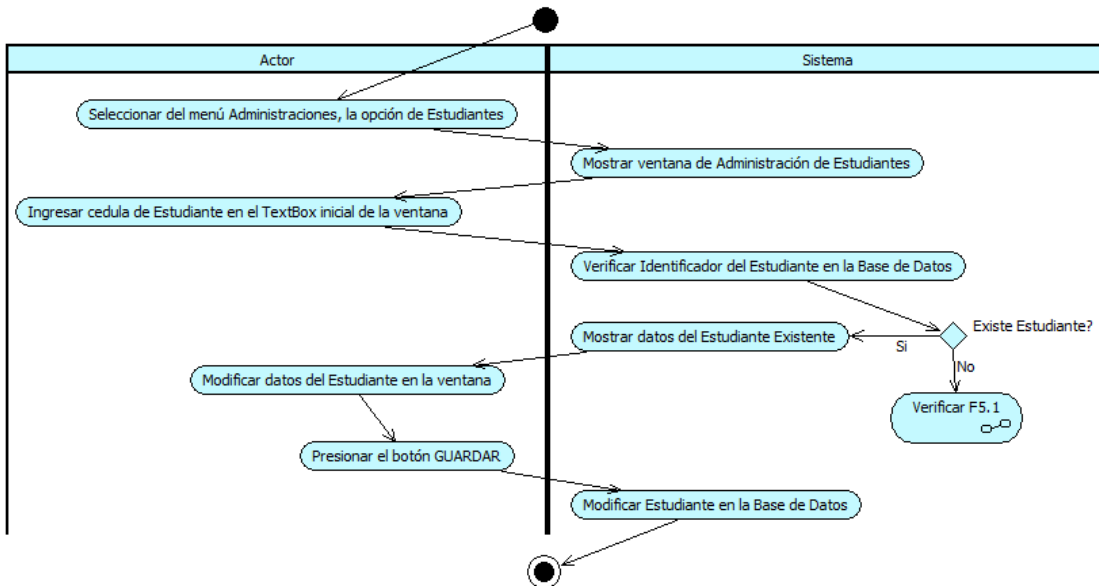


*Manejo de Estudiantes*

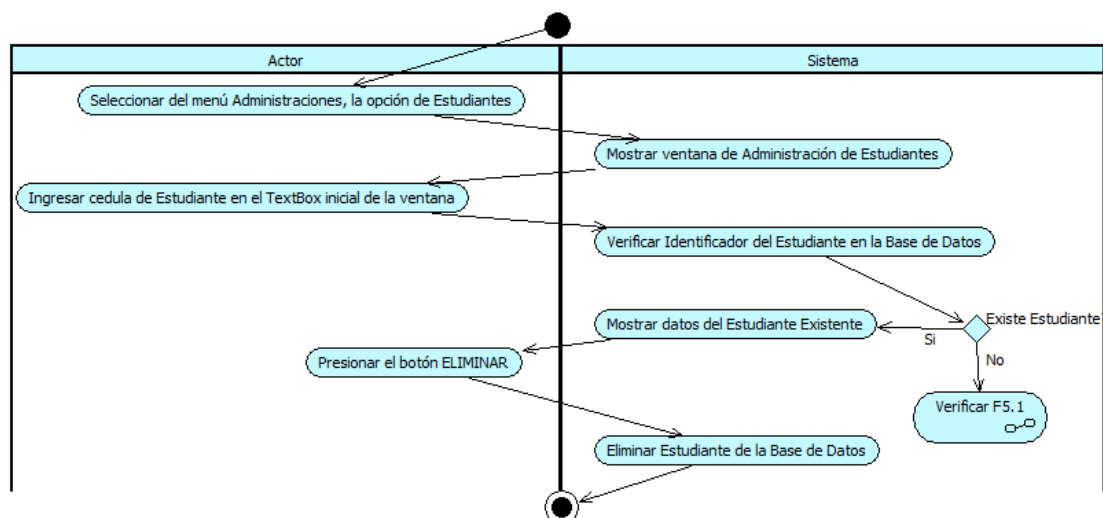
**F5.1. Ingreso de Estudiantes**



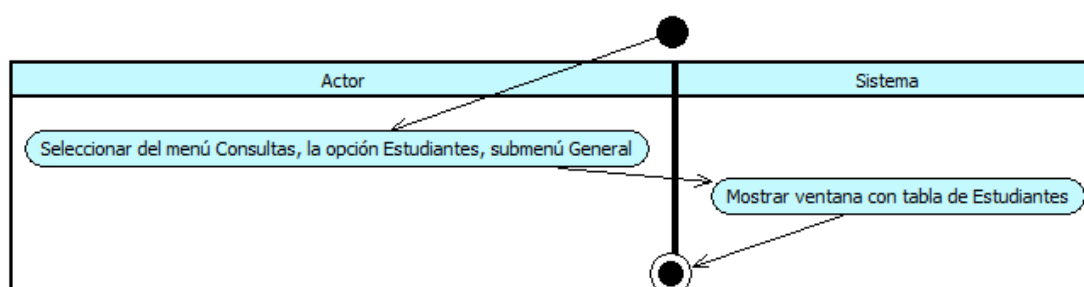
**F5.2. Modificación de Estudiantes**



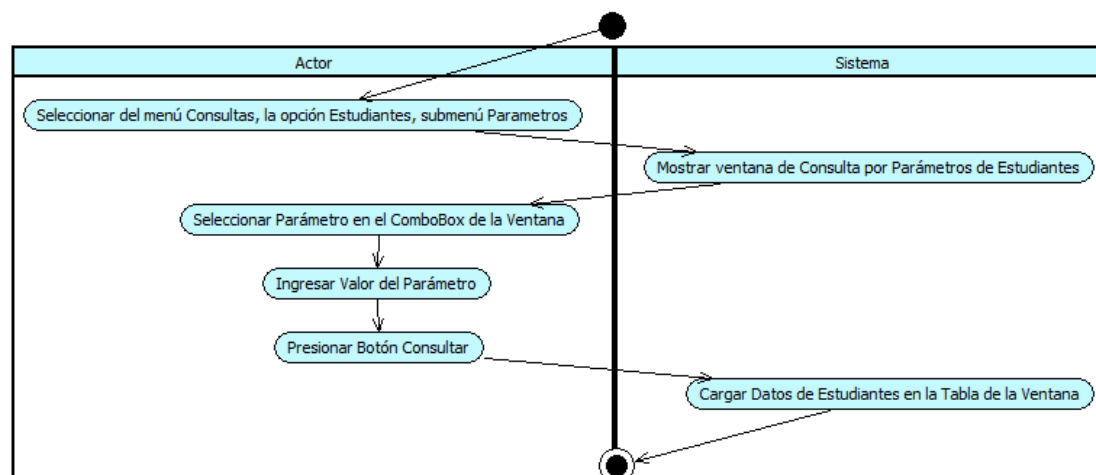
### F5.3. Eliminación de Estudiantes



### F7.5.1. Consulta general de Estudiantes

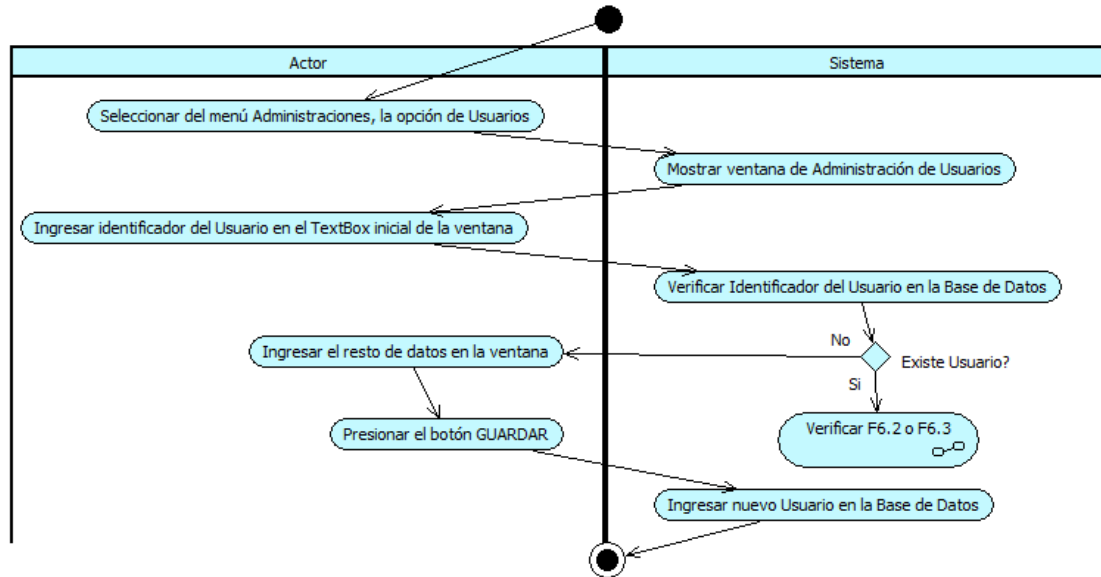


### F7.5.2. Consulta por Parámetros de Estudiantes

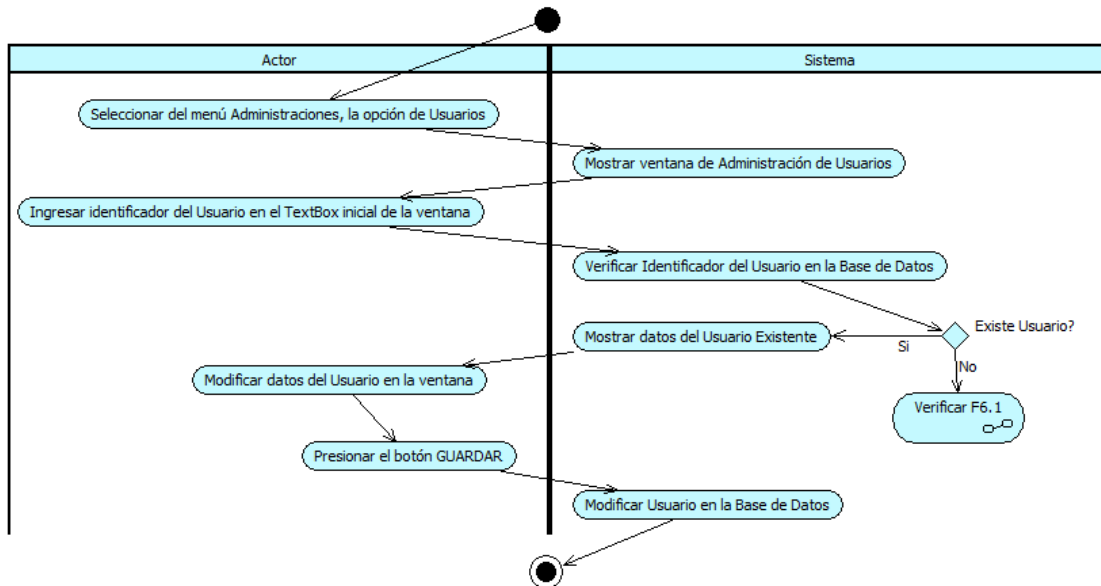


## Manejo de Usuarios

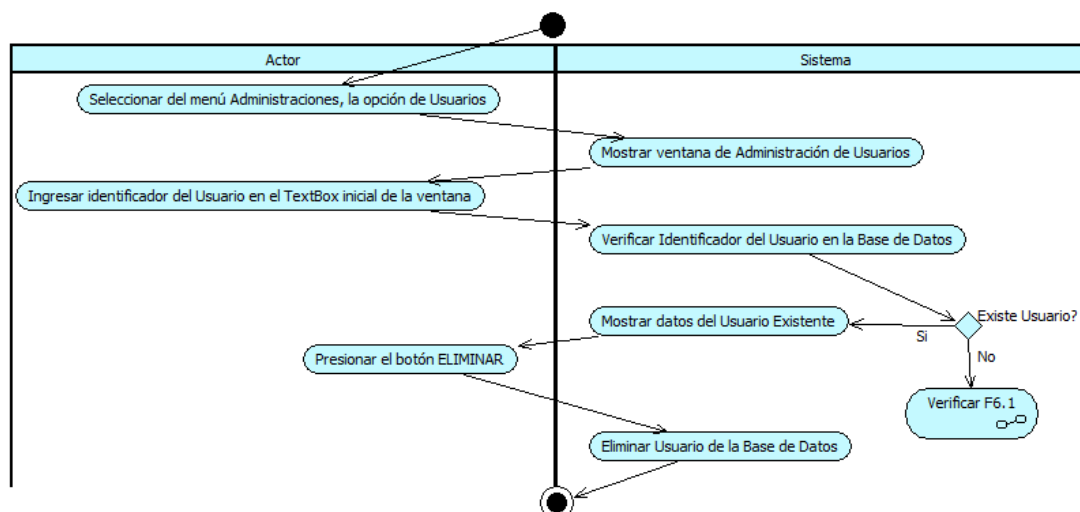
### F6.1. Ingreso de Usuarios



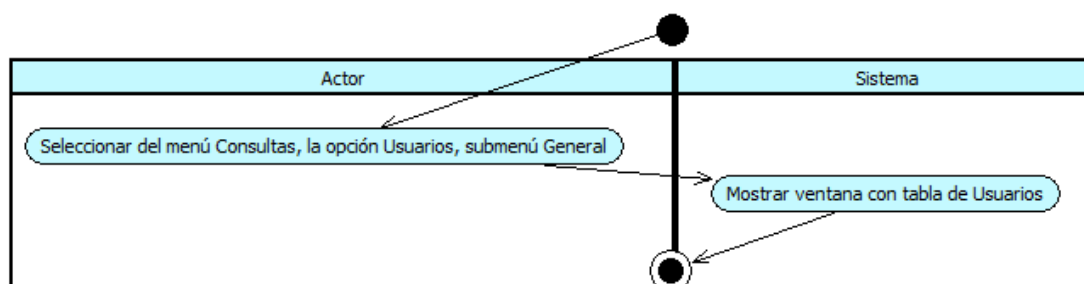
### F6.2. Modificación de Usuarios



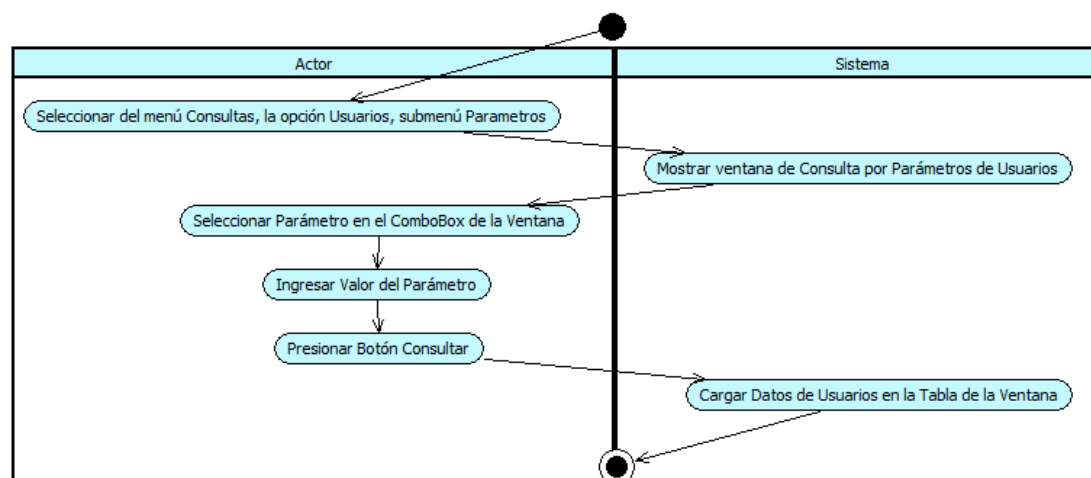
### F6.3. Eliminación de Usuarios



### F7.6.1. Consulta general de Usuarios



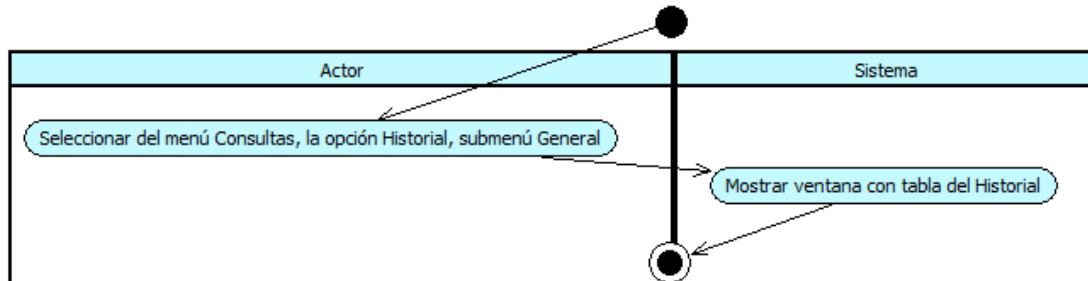
### F7.6.2. Consulta por Parámetros de Usuarios



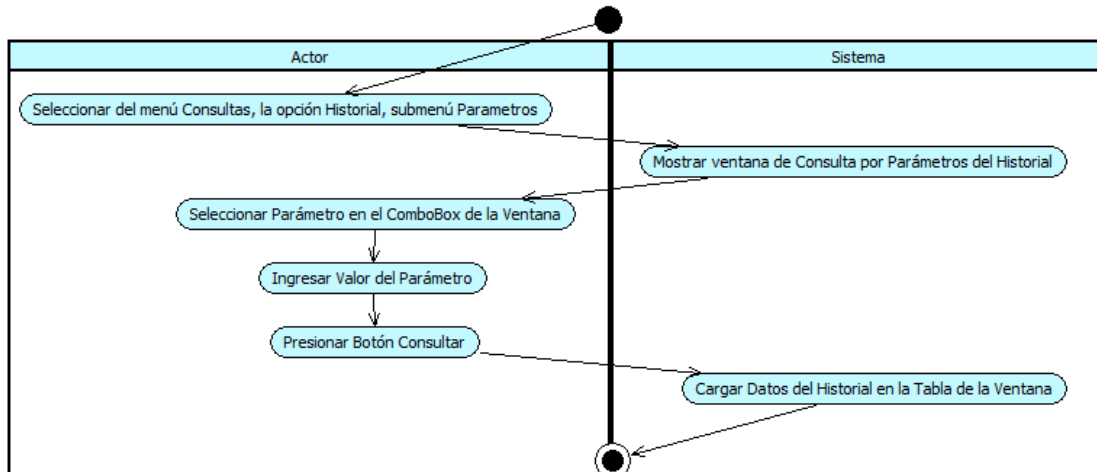


### *Consulta de Historial*

#### **F7.7.1. Consulta general del Historial**

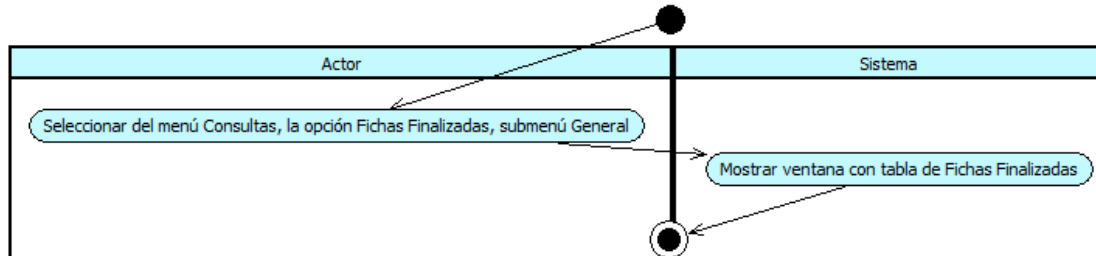


#### **F7.7.2. Consulta por Parámetros del Historial**

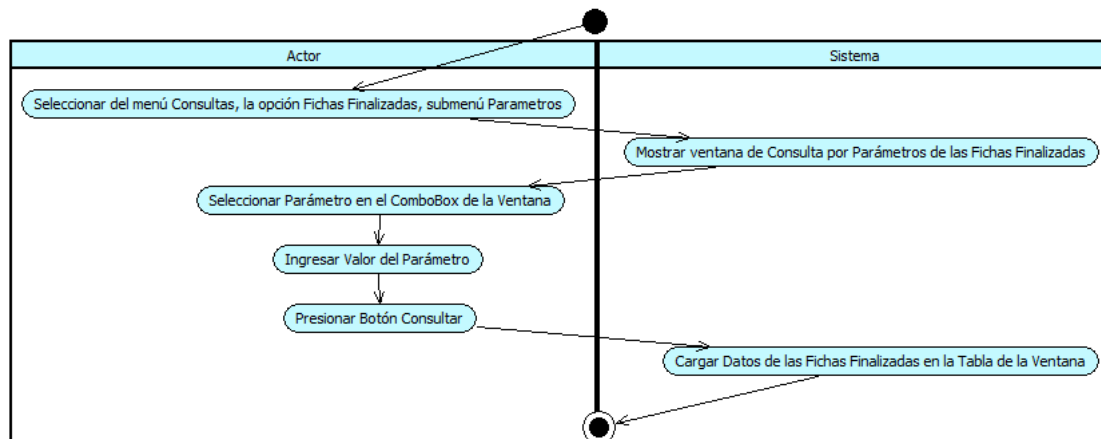


*Consulta de Fichas Finalizadas*

**F7.8.1. Consulta general de Fichas Finalizadas**



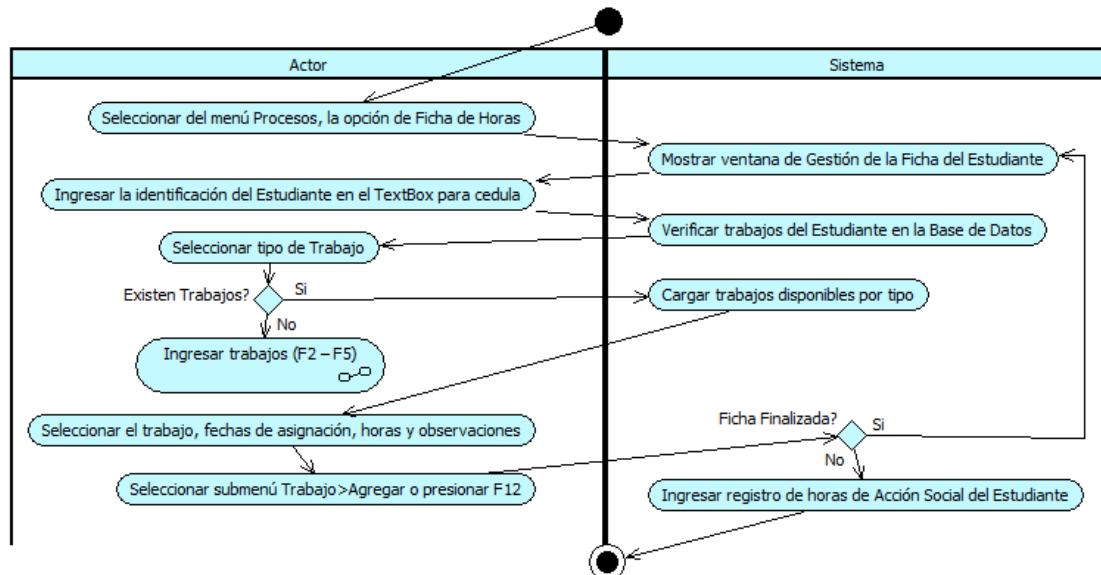
**F7.8.2. Consulta por Parámetros de Fichas Finalizadas**



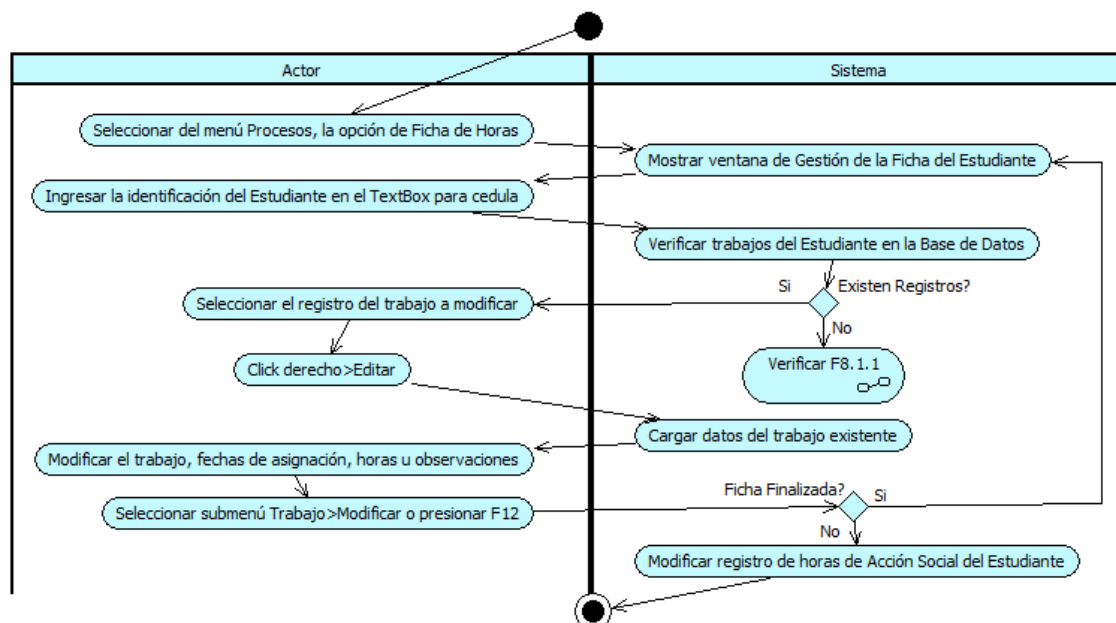
## F8. Gestión de fichas de acción social

### F8.1. Administración de horas de trabajos

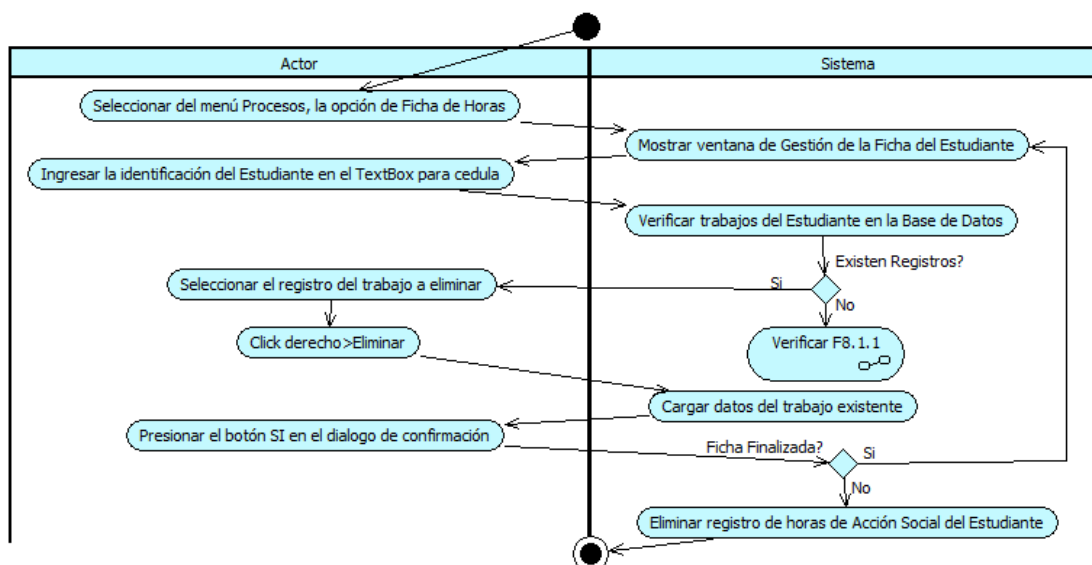
#### F8.1.1. Ingreso de Horas



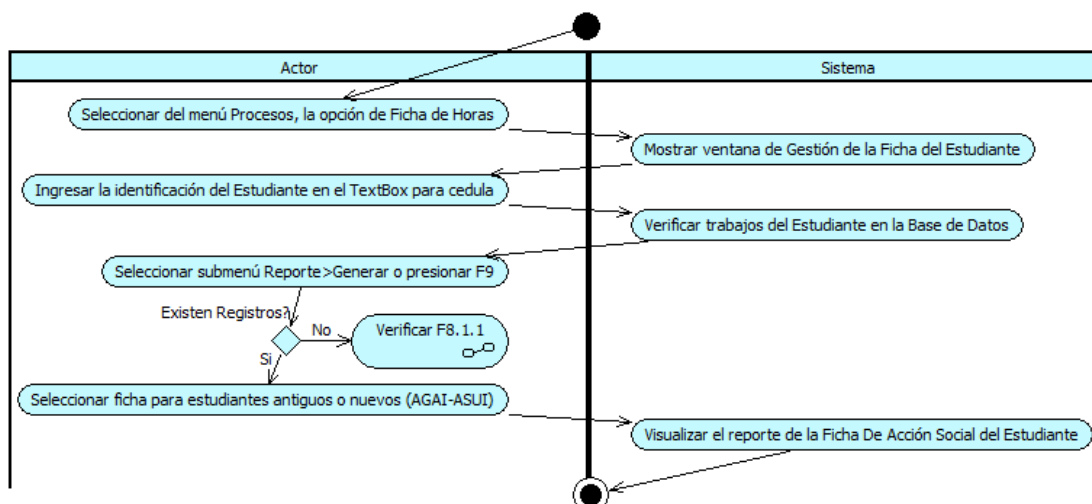
#### F8.1.2. Modificación de Horas



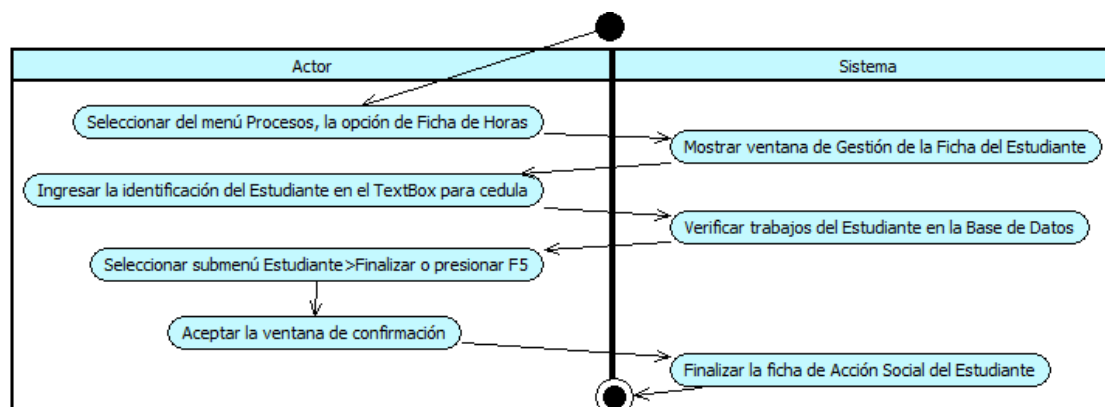
### F8.1.3. Eliminación de Horas



### F8.2. Generar Reporte de Ficha Estudiantil

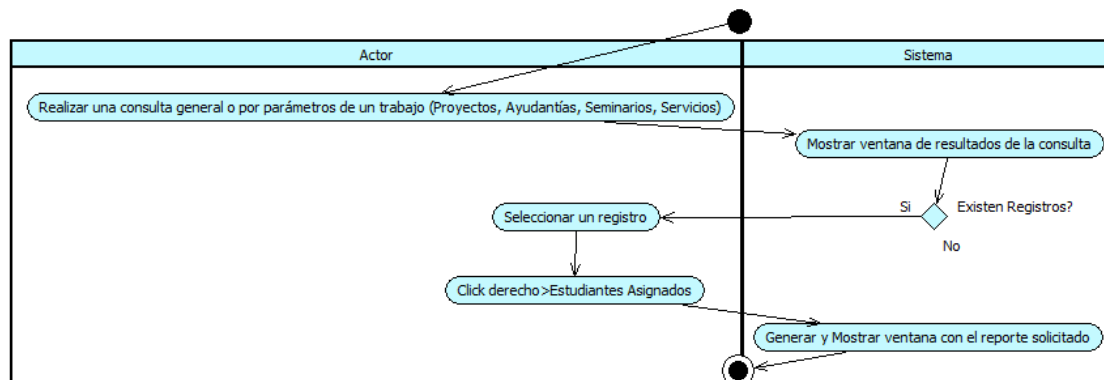


### F8.3. Finalizar Ficha Estudiantil

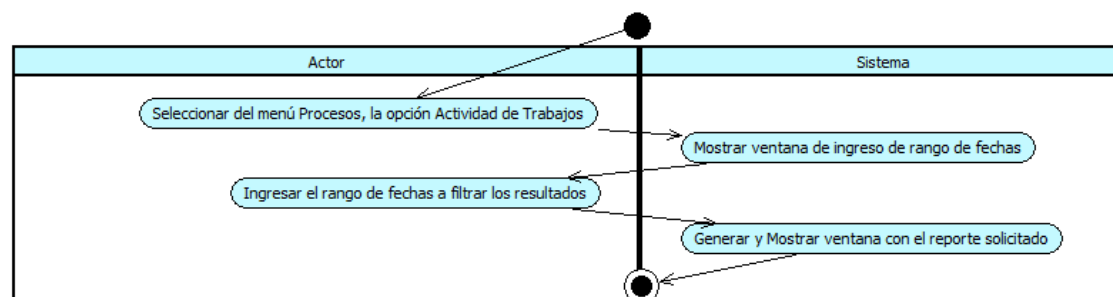


### *F9. Generación de reportes de actividades*

#### **F9.1. Generar reportes de actividad por trabajo**

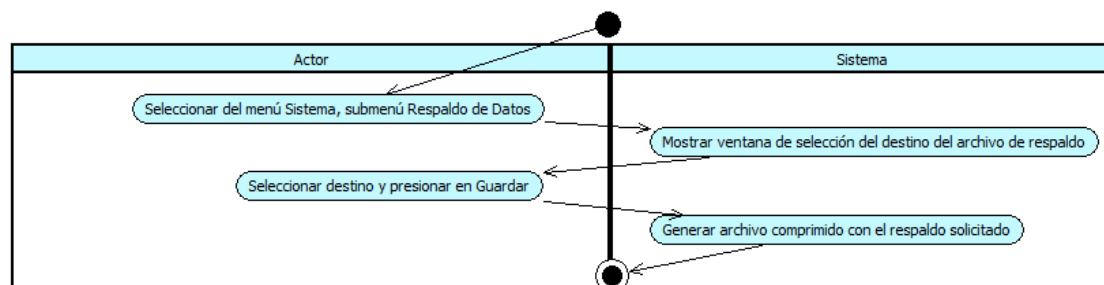


#### **F9.2. Generar reportes de actividad por fechas**



### *F10. Creación de respaldos de datos*

#### **F10. Crear respaldos de datos**



#### **5.1.4 Otros Requerimientos**

##### ***Requerimientos de Interfaces Externas***

##### **A nivel de Hardware (Requerimientos Mínimos)**

ORDENADOR	
Procesador	1,6 GHz
Memoria	512 MB
Disco Duro	30 GB

##### **A nivel de Software (Requerimientos Mínimos)**

ORDENADOR	
Sistema Operativo	Independiente
Java Virtual Machine	1.5

##### ***Restricciones de diseño e implementación***

##### **Lenguaje de Programación**

*JDK 1.5 o mayor*

##### **Base de Datos**

*MySQL 5.0 o mayor*

##### ***Requerimientos no funcionales***

Son aquellos requerimientos que no se refieren directamente a las funciones específicas que entrega el sistema, sino a las propiedades emergentes de éste como la fiabilidad, la respuesta en el tiempo y la capacidad de almacenamiento. De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y la representación de datos que se utiliza en la interface del sistema.

Muchos requerimientos no funcionales se refieren al sistema como un todo más que a rasgos particulares del mismo. Esto significa que a menudo son más críticos que los requerimientos funcionales particulares. Mientras que el incumplimiento de este último degradará el sistema, una falla en un requerimiento no funcional del sistema lo inutiliza.

Los requerimientos no funcionales surgen de la necesidad del usuario, debido a las restricciones en el presupuesto, a las políticas de la organización, a la necesidad de interoperabilidad con otros sistemas de software o hardware o a factores externos como los reglamentos de seguridad, las políticas de privacidad, etc.

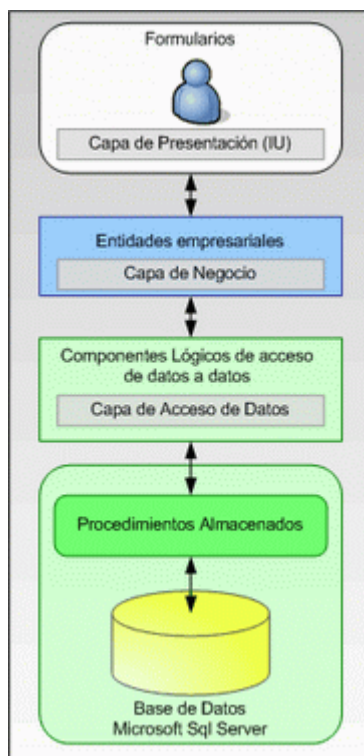
El sistema dependerá de los siguientes requerimientos no funcionales:

- N1.** El sistema deberá ser hecho bajo herramientas libres
- N2.** El sistema debe acoplarse a los requerimientos de hardware del área
- N3.** El sistema debe cumplir normas de desarrollo de la Facultad de Ingeniería
- N4.** El sistema debe manejar la base de datos de estudiantes de la Facultad

### 5.1.5 Estructura del Producto

#### Diseño Arquitectónico

En el diseño de sistemas informáticos actual se suele usar las arquitecturas multinivel o Programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).



#### Capas y niveles

**1. Capa de presentación:** es la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario.

**2. Capa de negocio:** es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las

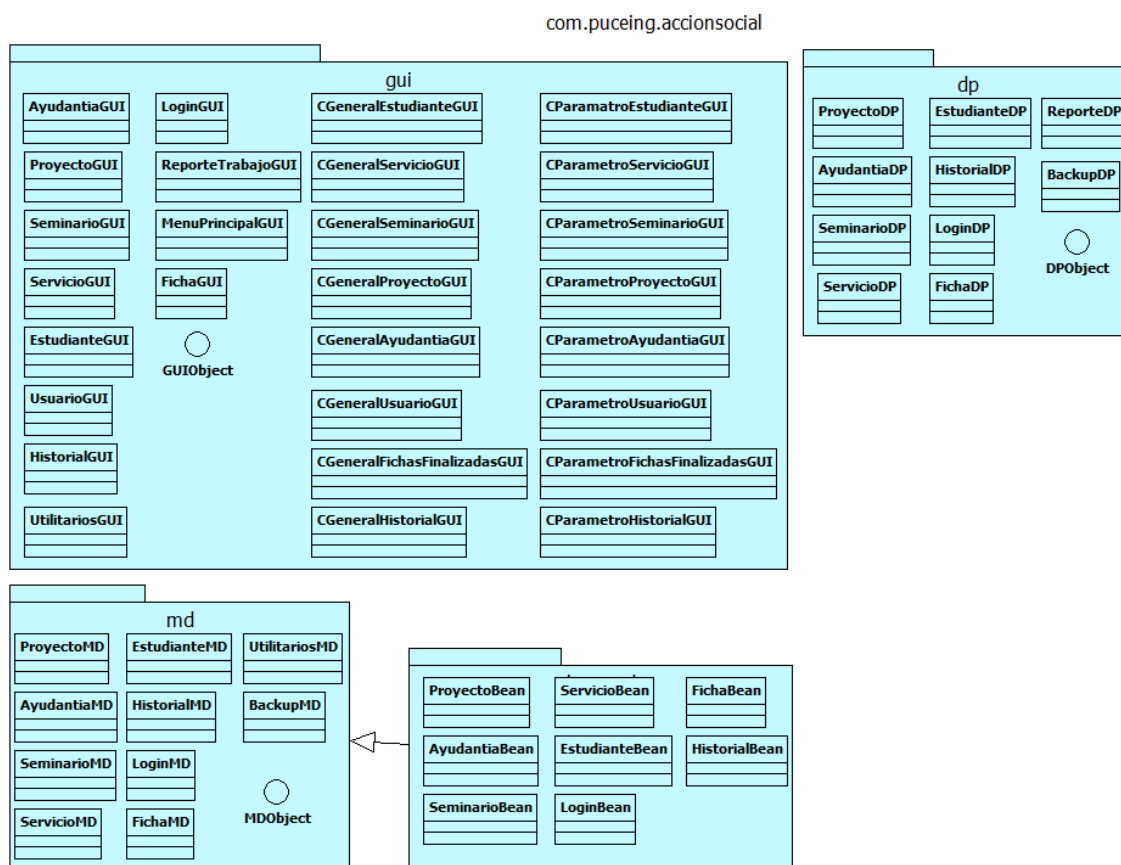


## IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

**3. Capa de datos:** es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

### *Diagrama de Paquetes*





**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

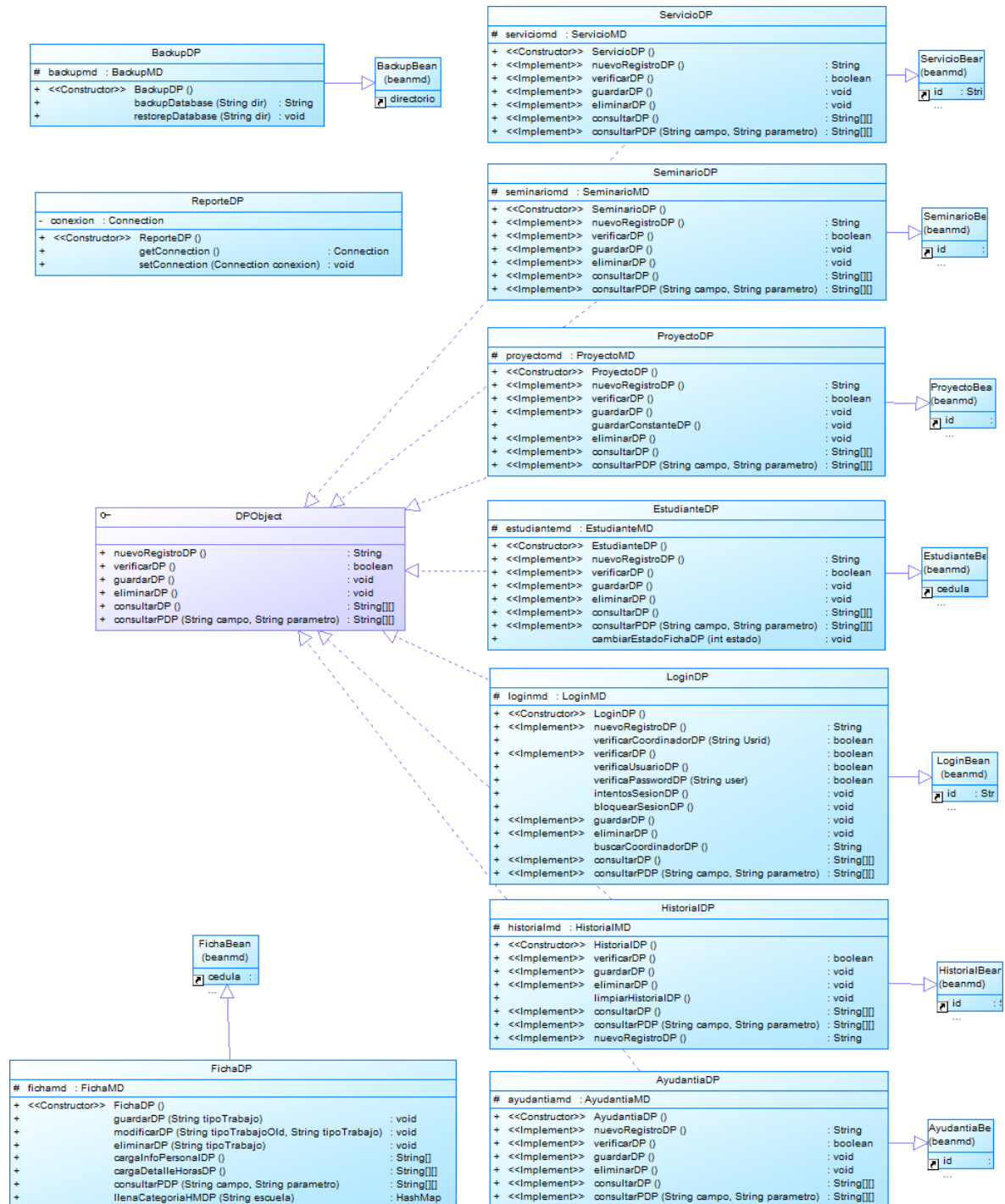
[illegible][illegible][illegible][illegible][illegible][illegible][illegible]

CensusDataIO	
sampleFrame	Panel
columns	String
data	String
data1	String
data2	String
data3	String
data4	String
data5	String
data6	String
data7	String
data8	String
data9	String
data10	String
data11	String
data12	String
data13	String
data14	String
data15	String
data16	String
data17	String
data18	String
data19	String
data20	String
data21	String
data22	String
data23	String
data24	String
data25	String
data26	String
data27	String
data28	String
data29	String
data30	String
data31	String
data32	String
data33	String
data34	String
data35	String
data36	String
data37	String
data38	String
data39	String
data40	String
data41	String
data42	String
data43	String
data44	String
data45	String
data46	String
data47	String
data48	String
data49	String
data50	String
data51	String
data52	String
data53	String
data54	String
data55	String
data56	String
data57	String
data58	String
data59	String
data60	String
data61	String
data62	String
data63	String
data64	String
data65	String
data66	String
data67	String
data68	String
data69	String
data70	String
data71	String
data72	String
data73	String
data74	String
data75	String
data76	String
data77	String
data78	String
data79	String
data80	String
data81	String
data82	String
data83	String
data84	String
data85	String
data86	String
data87	String
data88	String
data89	String
data90	String
data91	String
data92	String
data93	String
data94	String
data95	String
data96	String
data97	String
data98	String
data99	String
data100	String
data101	String
data102	String
data103	String
data104	String
data105	String
data106	String
data107	String
data108	String
data109	String
data110	String
data111	String
data112	String
data113	String
data114	String
data115	String
data116	String
data117	String
data118	String
data119	String
data120	String
data121	String
data122	String
data123	String
data124	String
data125	String
data126	String
data127	String
data128	String
data129	String
data130	String
data131	String
data132	String
data133	String
data134	String
data135	String
data136	String
data137	String
data138	String
data139	String
data140	String
data141	String
data142	String
data143	String
data144	String
data145	String
data146	String
data147	String
data148	String
data149	String
data150	String
data151	String
data152	String
data153	String
data154	String
data155	String
data156	String
data157	String
data158	String
data159	String
data160	String
data161	String
data162	String
data163	String
data164	String
data165	String
data166	String
data167	String
data168	String
data169	String
data170	String
data171	String

[illegible][illegible][illegible][illegible][illegible][illegible][illegible][illegible]

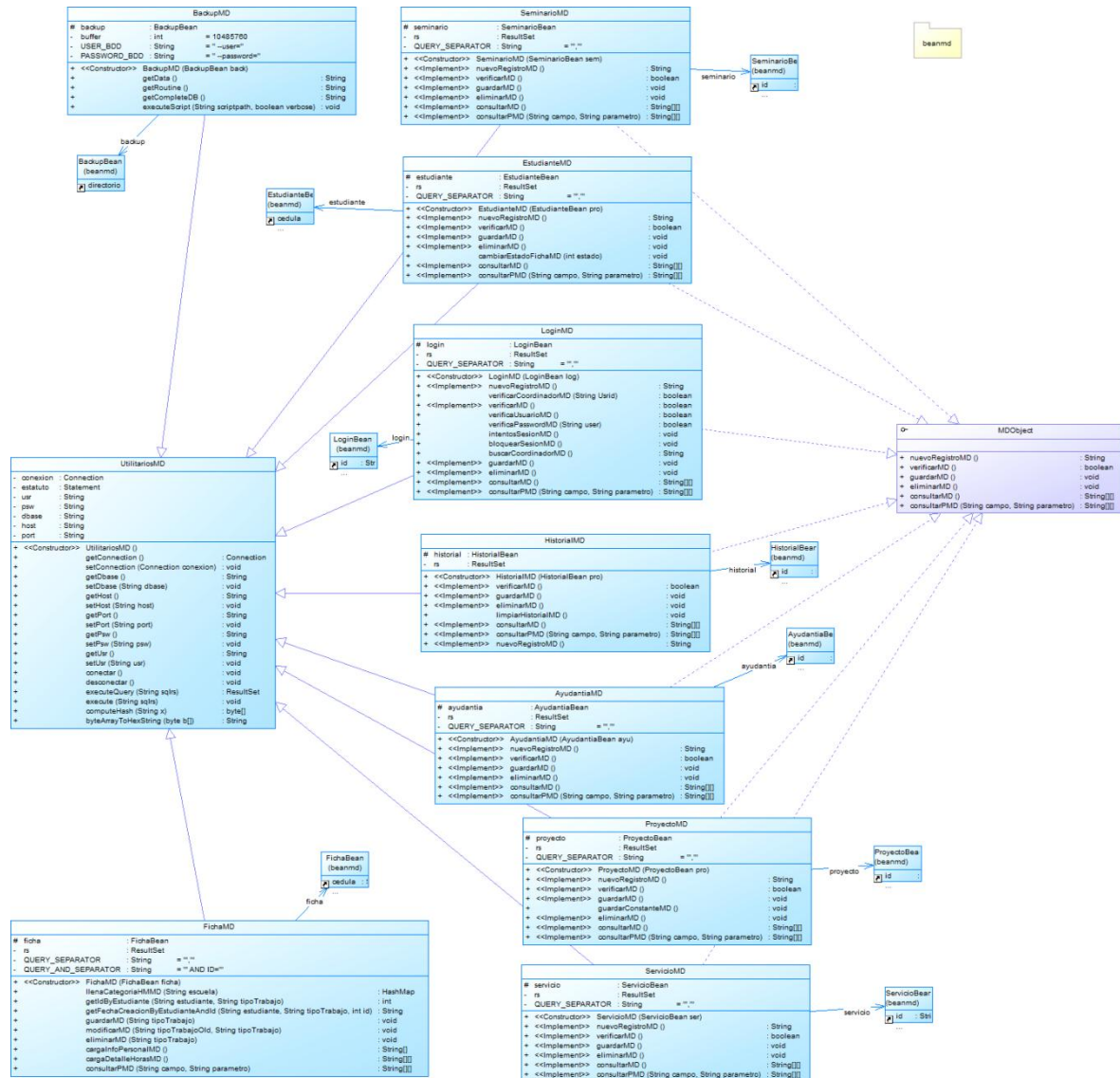
# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

## ■ Clases de Dominio del Problema (DP)



# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

## ■ Clases de Manejo de Datos (MD)





# IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

AyudantiaBean		
- id	: String	
- materia	: String	
- profesor	: String	
- tipo	: String	
- descripcion	: String	
- loginActualizacion	: String	
- fechaCreacion	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> AyudantiaBean ()		
+	setId (String id)	: void
+	setMateria (String materia)	: void
+	setProfesor (String profesor)	: void
+	setTipo (String tipo)	: void
+	setDescription (String descripcion)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
+	setFechaCreacion ()	: void
+	getId ()	: String
+	getMateria ()	: String
+	getTipo ()	: String
+	getProfesor ()	: String
+	getDescription ()	: String
+	getLoginActualizacion ()	: String
+	getFechaCreacion ()	: String
<<Getter>>		

ServicioBean		
- id	: String	
- tipo	: String	
- institucion	: String	
- descripcion	: String	
- fecha	: String	
- loginActualizacion	: String	
- fechaCreacion	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> ServicioBean ()		
+	setId (String id)	: void
+	setTipo (String tipo)	: void
+	setInstitucion (String institucion)	: void
+	setDescription (String descripcion)	: void
+	setFecha (String fecha)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
+	setFechaCreacion ()	: void
+	getId ()	: String
+	getTipo ()	: String
+	getInstitucion ()	: String
+	getDescription ()	: String
+	getFecha ()	: String
+	getLoginActualizacion ()	: String
+	getFechaCreacion ()	: String
<<Getter>>		

SeminarioBean		
- id	: String	
- numero	: int	
- tema	: String	
- expositor	: String	
- fecha	: String	
- hora	: String	
- duracion	: String	
- lugar	: String	
- loginActualizacion	: String	
- fechaCreacion	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> SeminarioBean ()		
+	setId (String id)	: void
+	setNumero (int numero)	: void
+	setTema (String tema)	: void
+	setExpositor (String expositor)	: void
+	setFecha (String fecha)	: void
+	setHora (String hora)	: void
+	setDuracion (String duracion)	: void
+	setLugar (String lugar)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
+	setFechaCreacion ()	: void
<<Getter>>		
+	getId ()	: String
+	getNumero ()	: int
+	getTema ()	: String
+	getExpositor ()	: String
+	getFecha ()	: String
+	getHora ()	: String
+	getDuracion ()	: String
+	getLugar ()	: String
+	getLoginActualizacion ()	: String
+	getFechaCreacion ()	: String
<<Getter>>		

HistorialBean		
- id	: String	
- evento	: String	
- loginActualizacion	: String	
- fechaActualizacion	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> HistorialBean ()		
+	setId (String id)	: void
+	setEvento (String evento)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
+	setFechaActualizacion ()	: void
+	setFechaActualizacion (String fechaActualizacion)	: void
+	getId ()	: String
+	getEvento ()	: String
+	getLoginActualizacion ()	: String
+	getFechaActualizacion ()	: String
<<Getter>>		

FichaBean		
- ocdula	: String	
- nivel	: int	
- fecha_asignacion	: String	
- trabajo	: String	
- trabajoOld	: String	
- fecha_entrega	: String	
- horas	: int	
- observaciones	: String	
- loginActualizacion	: String	
- fechaCreacion	: String	
- id	: int	
- estudiante	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> FichaBean ()		
+	setId (int id)	: void
+	setEstudiante (String estudiante)	: void
+	setOcdula (String ocdula)	: void
+	setNivel (int nivel)	: void
+	setFecha_asignacion (String fecha_asignacion)	: void
+	setTrabajo (String trabajo)	: void
+	setTrabajoOld (String trabajoOld)	: void
+	setFecha_entrega (String fecha_entrega)	: void
+	setHoras (int horas)	: void
+	setObservaciones (String observaciones)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
<<Setter>>		
+	setFechaCreacion ()	: void
+	getId ()	: void
+	getEstudiante ()	: String
+	getOcdula ()	: String
+	getNivel ()	: int
+	getFecha_asignacion ()	: String
+	getTrabajo ()	: String
+	getTrabajoOld ()	: String
+	getFecha_entrega ()	: String
+	getHoras ()	: int
+	getObservaciones ()	: String
+	getLoginActualizacion ()	: String
+	getFechaCreacion ()	: String
<<Getter>>		

EstudianteBean		
- ocdula	: String	
- apellidos	: String	
- nombres	: String	
- nivel	: int	
- matricula	: char	
- direccion	: String	
- telefono	: String	
- celular	: String	
- email	: String	
- escuela	: String	
- loginActualizacion	: String	
- fechaCreacion	: String	
- estado	: int	
- id	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> EstudianteBean ()		
+	setId (String id)	: void
+	setOcdula (String ocdula)	: void
+	setApellidos (String apellidos)	: void
+	setNombres (String nombres)	: void
+	setMatricula (char matricula)	: void
+	setNivel (int nivel)	: void
+	setDireccion (String direccion)	: void
+	setTelefono (String telefono)	: void
+	setCelular (String celular)	: void
+	setEscuela (String escuela)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
<<Setter>>		
+	setFechaCreacion ()	: void
+	setEmail (String email)	: void
+	setEstado (int estado)	: void
<<Getter>>		
+	getId ()	: String
+	getOcdula ()	: String
+	getApellidos ()	: String
+	getNombres ()	: String
+	getNivel ()	: int
+	getMatricula ()	: char
+	getDireccion ()	: String
+	getTelefono ()	: String
+	getCelular ()	: String
+	getEscuela ()	: String
+	getLoginActualizacion ()	: String
+	getFechaCreacion ()	: String
+	getEmail ()	: String
+	getEstado ()	: int

LoginBean		
- id	: String	
- user	: String	
- password	: String	
- status	: String	
- intentos	: int	
- tipo	: String	
- database	: String	
- loginActualizacion	: String	
- fechaCreacion	: String	
- nombre	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> LoginBean ()		
+	setId (String id)	: void
+	setUser (String user)	: void
+	setPassword (String password)	: void
+	setStatus (String status)	: void
+	setTipo (String tipo)	: void
+	setIntentos (int intentos)	: void
+	setDatabase (String database)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
<<Setter>>		
+	setFechaCreacion ()	: void
+	setNombre (String nombre)	: void
+	getId ()	: String
+	getUser ()	: String
+	getPassword ()	: String
+	getStatus ()	: String
+	getTipo ()	: String
+	getIntentos ()	: int
+	getDatabase ()	: String
+	getLoginActualizacion ()	: String
+	getFechaCreacion ()	: String
<<Getter>>		

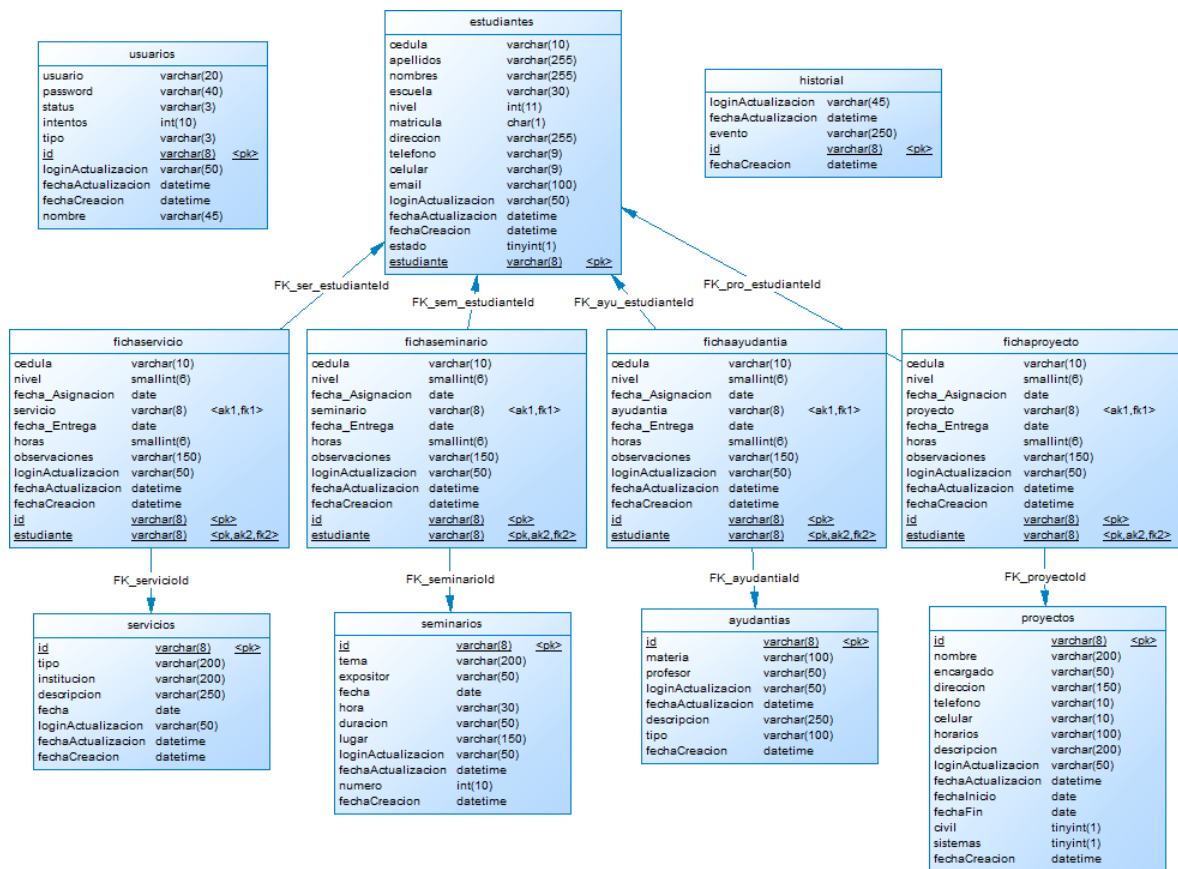
BackupBean		
- directorio	: String	
+ <<Constructor>> BackupBean ()		
+	getDirectorio ()	: String
<<Getter>>		

ProyectoBean		
- id	: String	
- nombre	: String	
- encargado	: String	
- direccion	: String	
- telefono	: String	
- celular	: String	
- fechaInicio	: String	
- fechaFin	: String	
- horarios	: String	
- descripcion	: String	
- tipo	: int	
- sistemas	: int	
- loginActualizacion	: String	
- fechaCreacion	: String	
- formatoFecha	: SimpleDateFormat	
+ <<Constructor>> ProyectoBean ()		
+	setId (String id)	: void
+	setNombre (String nombre)	: void
+	setEncargado (String encargado)	: void
+	setDireccion (String direccion)	: void
+	setTelefono (String telefono)	: void
+	setCelular (String celular)	: void
+	setDescription (String descripcion)	: void
+	setLoginActualizacion (String loginActualizacion)	: void
<<Setter>>		
+	setFechaCreacion ()	: void
+	setHorarios (String horarios)	: void
+	setFechaInicio (String fechaInicio)	: void
+	setFechaFin (String fechaFin)	: void
+	setCivil (int civil)	: void
+	setSistemas (int sistemas)	: void
<<Getter>>		
+	getId ()	: String
+	getNombre ()	: String
+	getEncargado ()	: String
+	getDireccion ()	: String
+	getTelefono ()	: String
+	getCelular ()	: String
+	getDescription ()	: String
+	getLoginActualizacion ()	: String
+	getFechaCreacion ()	: String
+	getHorarios ()	: String
+	getFechaInicio ()	: String
+	getFechaFin ()	: String
+	getCivil ()	: int
+	getSistemas ()	: int



## IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE

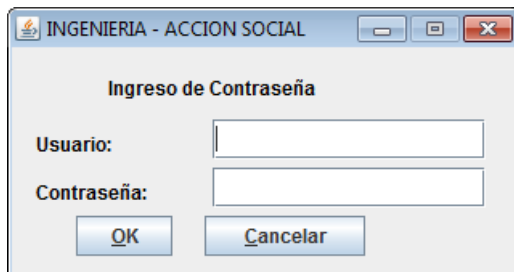
### ■ Diagrama Físico (E / R)



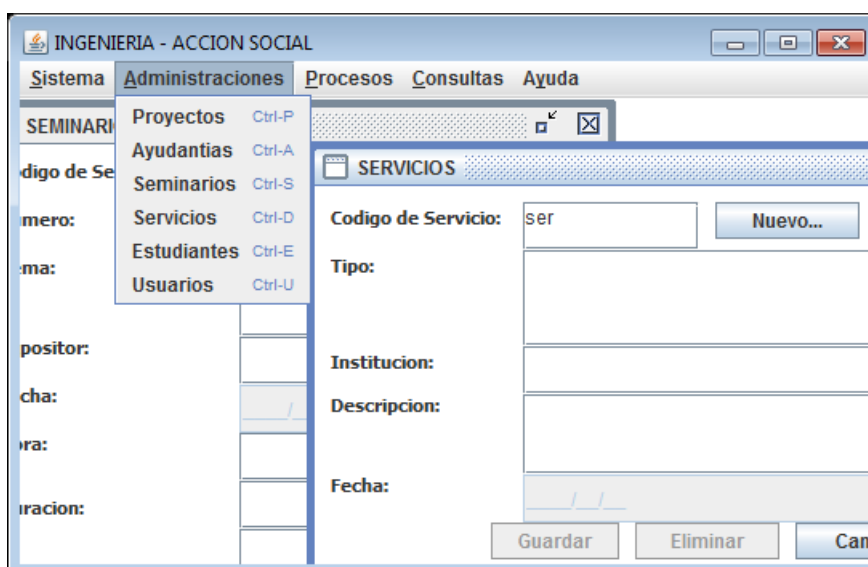
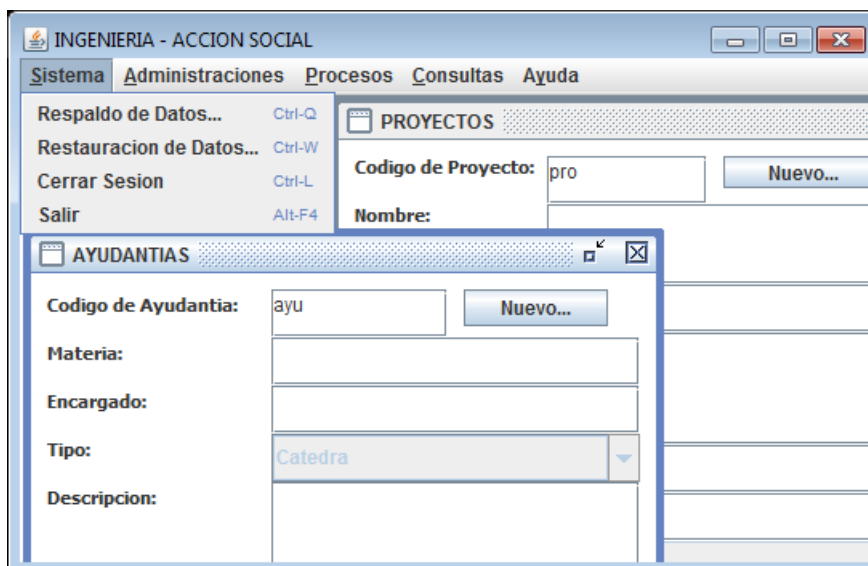
## 5.2Elaboración

### 5.2.1 Prototipo Ejecutable

#### *Ventana de Ingreso (Login)*

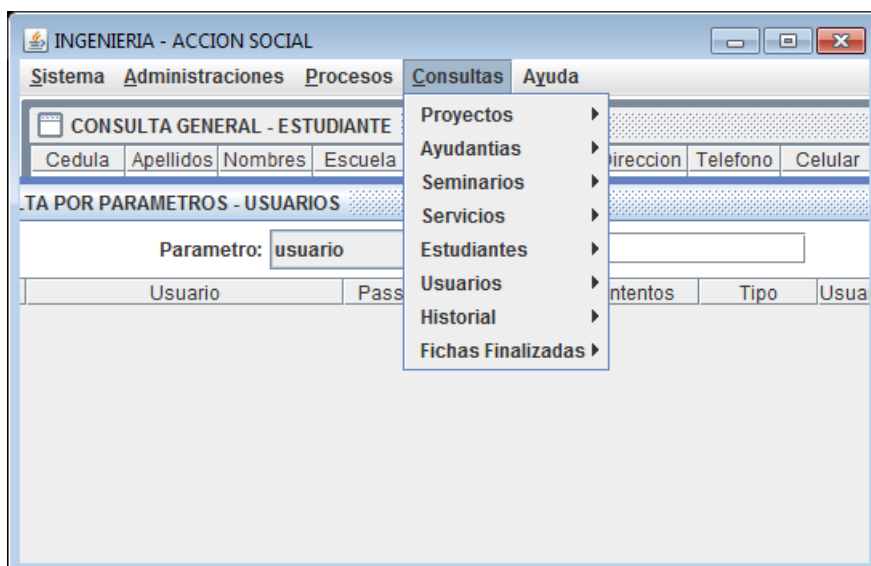
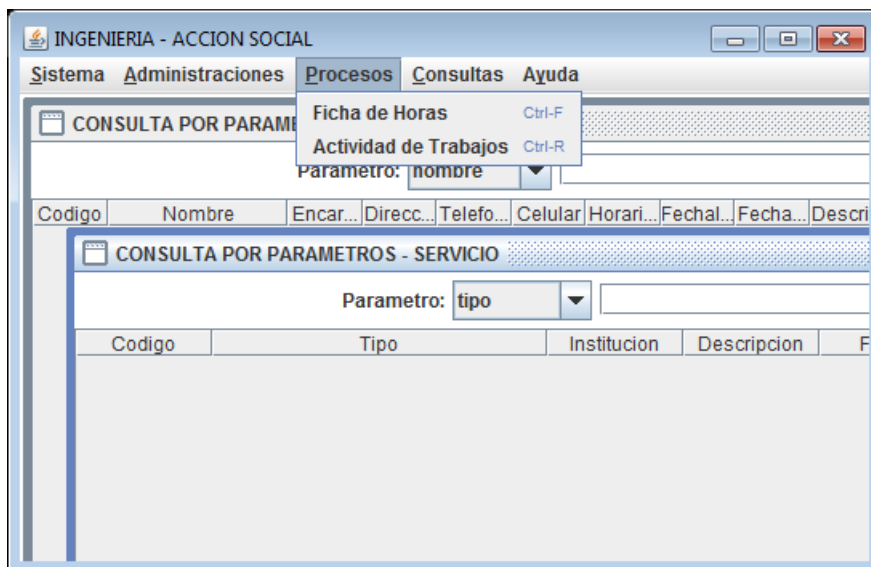


#### *Ventana Principal*

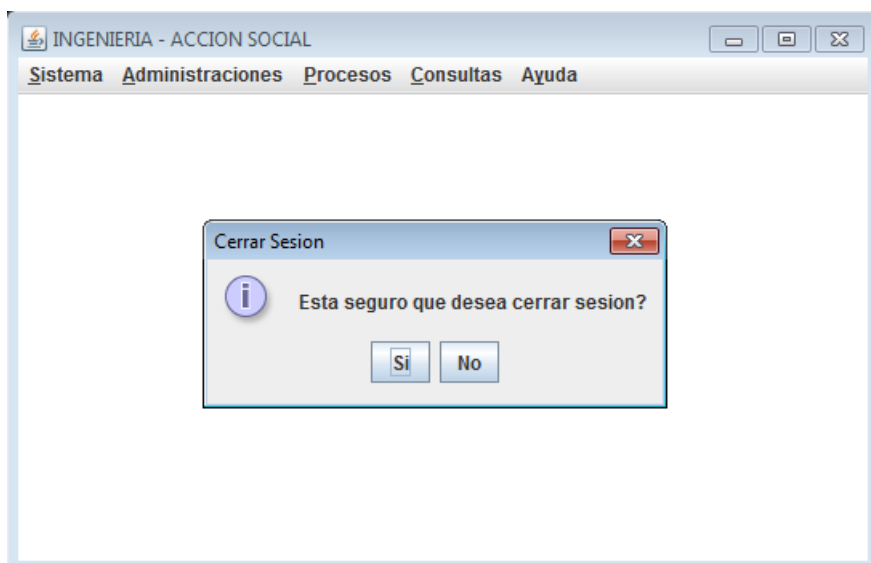




## IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA FACULTAD DE INGENIERIA DE LA PUCE



### *Ventanas de Diálogo*



### 5.2.2 *Lista de Riesgos*

En todo modelo de desarrollo de software existen varias circunstancias que puedan implicar riesgos para el proyecto, afectando los tiempos de entrega o hasta las capacidades del sistema.

Es importante tener claros los posibles riesgos que se pueden presentar al momento de desarrollar la aplicación, esto ayuda a saber con tiempo los caminos a tomar y llegar a una solución rápida sin afectar el tiempo de vida del proyecto.

Para el desarrollo de la aplicación se tienen claros los siguientes riesgos:

- ***Tiempo de desarrollo muy corto, comparado con las capacidades del sistema.***

En este caso, hay que llevar un control diario con el cliente y hacerle constar el avance del proyecto en cada reunión, charlar sobre los problemas y fijar posibles extensiones de tiempo.

- ***Limitación de funcionalidades debido a las herramientas seleccionadas.***

Se debe armar un listado con las características, ventajas y desventajas de cada herramienta a usar. En caso de encontrar una característica que pueda parar el tiempo de entrega, complementar con una solución alternativa, como una herramienta del mismo propósito, pero de diferente desarrollador.

- ***Pérdida de información de los módulos a desarrollar, debido a daños de hardware.***

Crear respaldos periódicos de las fuentes utilizadas o bien, almacenar la información en un servidor de repositorios, como subversión o mercurial<sup>1</sup>.

---

<sup>1</sup> Herramientas usadas para control y versionamiento de fuentes. Revisar bibliografía capítulo 5

## 5.3 Construcción

### 5.3.1 Construcción

La construcción del sistema se facilitó, debido a la previa estructuración y diseño mediante las herramientas de modelado UML.

Se partió de un prototipo genérico ejecutable previamente desarrollado, que consta de 4 administraciones básicas con sus consultas. Además de un proceso que implementa estas administraciones.

En base a este prototipo, se adaptaron las clases y entidades para que se sincronicen con los datos manipulados desde el modelo de base de datos.

El código fuente del aplicativo, se encuentra en el disco óptico entregado con el presente documento.

### 5.3.2 Plan de pruebas del sistema

#### Caso de Prueba: F1. Administración de Proyectos

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Administración de Proyectos</b>	Desplegar plantilla de Proyectos	F1
<b>Ingreso de Datos de Proyecto</b>		F1.1
<ul style="list-style-type: none"> <li><b>Correctos</b></li> <li><b>Incorrectos</b></li> </ul>	<p>Información se almacena</p> <p>Mensaje de Error apropiado</p>	
<b>Consulta de Proyectos</b>		
<ul style="list-style-type: none"> <li><b>General</b></li> <li><b>Parámetros</b></li> </ul>	<p>Mostrar información apropiada</p> <p>Presentar información filtrada</p>	<p>F7.1.1</p> <p>F7.1.2</p>
<b>Modificación de Datos de Proyecto</b>		F1.2
<ul style="list-style-type: none"> <li><b>Correctos</b></li> </ul>	Información se modifica	

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Eliminación de Proyecto</b>	Información Eliminada	F1.3

**Caso de Prueba: F2. Administración de Ayudantías**

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Administración de Ayudantías</b>	Desplegar plantilla de Ayudantías	F2
<b>Ingreso de Datos de Ayudantía</b>		F2.1
• <b>Correctos</b>	Información se almacena	
• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Consulta de Ayudantías</b>		
• <b>General</b>	Mostrar información apropiada	F7.2.1
• <b>Parámetros</b>	Presentar información filtrada	F7.2.2
<b>Modificación de Datos de Ayudantía</b>		F2.2
• <b>Correctos</b>	Información se modifica	
• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Eliminación de Ayudantía</b>	Información Eliminada	F2.3

**Caso de Prueba: F3. Administración de Seminarios**

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Administración de Seminarios</b>	Desplegar plantilla de Seminarios	F3
<b>Ingreso de Datos de Seminario</b>		F3.1
• <b>Correctos</b>	Información se almacena	



**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Consulta de Seminarios</b>		
• <b>General</b>	Mostrar información apropiada	F7.3.1
• <b>Parámetros</b>	Presentar información filtrada	F7.3.2
<b>Modificación de Datos de Seminario</b>		F3.2
• <b>Correctos</b>	Información se modifica	
• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Eliminación de Seminario</b>	Información Eliminada	F3.3

***Caso de Prueba: F4. Administración de Servicios***

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Administración de Servicios</b>	Desplegar plantilla de Servicios	F4
<b>Ingreso de Datos de Servicio</b>		F4.1
• <b>Correctos</b>	Información se almacena	
• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Consulta de Servicios</b>		
• <b>General</b>	Mostrar información apropiada	F7.4.1
• <b>Parámetros</b>	Presentar información filtrada	F7.4.2
<b>Modificación de Datos de Servicio</b>		F4.2
• <b>Correctos</b>	Información se modifica	
• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Eliminación de Servicio</b>	Información Eliminada	F4.3



*Caso de Prueba: F5. Administración de Estudiantes*

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Administración de Estudiantes</b>	Desplegar plantilla de Estudiantes	F5
<b>Ingreso de Datos de Estudiante</b>		F5.1
<ul style="list-style-type: none"> <li>• <b>Correctos</b></li> <li>• <b>Incorrectos</b></li> </ul>	<p>Información se almacena</p> <p>Mensaje de Error apropiado</p>	
<b>Consulta de Estudiantes</b>		
<ul style="list-style-type: none"> <li>• <b>General</b></li> <li>• <b>Parámetros</b></li> </ul>	<p>Mostrar información apropiada</p> <p>Presentar información filtrada</p>	<p>F7.5.1</p> <p>F7.5.2</p>
<b>Modificación de Datos de Estudiante</b>		F5.2
<ul style="list-style-type: none"> <li>• <b>Correctos</b></li> <li>• <b>Incorrectos</b></li> </ul>	<p>Información se modifica</p> <p>Mensaje de Error apropiado</p>	
<b>Eliminación de Estudiante</b>	Información Eliminada	F5.3

*Caso de Prueba: F6. Administración de Usuarios*

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Administración de Usuarios</b>	Desplegar plantilla de Usuarios	F6
<b>Ingreso de Datos de Usuario</b>		F6.1
<ul style="list-style-type: none"> <li>• <b>Correctos</b></li> <li>• <b>Incorrectos</b></li> </ul>	<p>Información se almacena</p> <p>Mensaje de Error apropiado</p>	
<b>Consulta de Usuarios</b>		

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

• <b>General</b>	Mostrar información apropiada	F7.6.1
• <b>Parámetros</b>	Presentar información filtrada	F7.6.2
<b>Modificación de Datos de Usuario</b>		F6.2
• <b>Correctos</b>	Información se modifica	
• <b>Incorrectos</b>	Mensaje de Error apropiado	
<b>Eliminación de Usuario</b>	Información Eliminada	F6.3

***Caso de Prueba: F8. Gestión de Fichas***

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Proceso Fichas de Horas</b>	Desplegar plantilla de Ficha de Horas	F8
<b>Ingreso de Datos de Horas</b>		
• <b>Correctos</b>	Información se almacena	F8.1.1
• <b>Incorrectos</b>	Mensaje de Error apropiado	
• <b>Ficha Finalizada</b>	Mensaje de Error apropiado	
<b>Modificación de Datos de Horas</b>		
• <b>Correctos</b>	Información se modifica	F8.1.2
• <b>Incorrectos</b>	Mensaje de Error apropiado	
• <b>Ficha Finalizada</b>	Mensaje de Error apropiado	
<b>Eliminación de Horas</b>		
• <b>Correcto</b>	Información Eliminada	F8.1.3
• <b>Ficha Finalizada</b>	Mensaje de Error apropiado	
<b>Generación de Reporte</b>		
• <b>Con datos</b>	Mostrar Reporte	F8.2

**IMPLEMENTACION DEL PROCESO DE GESTION DE HORAS DE EXTENSION UNIVERSITARIA PARA LA  
FACULTAD DE INGENIERIA DE LA PUCE**

• <b>Sin datos</b>	Mensaje de Error apropiado	
<b>Finalizar Ficha</b>		
• <b>Correcto</b>	Cambiar ficha a finalizada	F8.3
• <b>Incorrecto</b>	Mensaje de Error apropiado	

***Caso de Prueba: F9. Generación de Reportes de Actividades***

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Proceso Actividad de Trabajos o Consulta de Trabajos</b>	Desplegar plantilla actividades por fecha o por trabajo	F9
<b>Actividad de Trabajos</b>		
• <b>Fechas Correctas</b>	Mostrar Reporte	F9.1
• <b>Fechas Incorrectas</b>	Mensaje de Error apropiado	
<b>Consulta de Trabajos</b>		
• <b>Proyectos</b>	Mostrar Reporte	F9.2
• <b>Ayudantías</b>	Mostrar Reporte	F9.2
• <b>Seminarios</b>	Mostrar Reporte	F9.2
• <b>Servicios</b>	Mostrar Reporte	F9.2
• <b>No Trabajo</b>	Mostrar Reporte	F9.2
	Mensaje de Error apropiado	

***Caso de Prueba: F10. Creación de Respaldos***

Entradas	Resultados Esperados	Casos de Uso
<b>Seleccionar Sistema Respaldo de Datos</b>	Desplegar ventana de destino del archivo	F10
<b>Directorio destino</b>		
• <b>Correcto</b>	Guardar respaldo solicitado	F10
• <b>Incorrecto</b>	Mensaje de Error apropiado	





## 5.4 Transición

### 5.4.1 Seguridades

El sistema no maneja seguridad a nivel de bases de datos, ya que usa el mismo servidor de base de datos de la facultad de ingeniería para el manejo de estudiantes.

Se maneja una única instancia de la aplicación, ya que no pretende ser distribuida ni que genere sesiones por servidor.

Las contraseñas de usuario, para acceder al sistema se almacenarán en la base de datos tras un hash de una sola vía. Lo que quiere decir, que se generará un código hexadecimal a partir de la clave enviada, pero no se podrá obtener la clave original, teniendo el código hexadecimal previamente generado.

Los reportes generados tienen la opción de almacenarse en el disco duro o en alguna unidad de almacenamiento. En caso de no elegir un método de almacenamiento, estos se borrarán de memoria.

Los archivos de respaldos creados desde la aplicación contienen 3 scripts.

- Script de respaldo de la base de datos completa
- Script de respaldo solo de la información de las tablas
- Script de respaldo solo de las funciones y procedimientos del modelo

El usuario podrá almacenar estos archivos en cualquier unidad de almacenamiento.

En caso de requerir una recuperación de datos a partir de un archivo de respaldos, se debe coordinar con el desarrollador del sistema o bien algún personal de mantenimiento previamente entrenado.

### 5.4.2 Costos

El sistema no tendrá costo alguno, ya que es parte de un proyecto de tesis como sistema para la facultad y está construido bajo herramientas libres en su totalidad.

### **5.4.3 Despliegue**

El sistema, se instalará bajo las especificaciones dadas en los requerimientos no funcionales.

En caso de haber problemas con dichas especificaciones, se procederá a realizar una nueva lista de requerimientos no funcionales tanto en Hardware como Software para fijar una nueva fecha de entrega según el peso del cambio.

### **5.4.4 Mantenimiento**

Se ofrecerá un chequeo personal del sistema cada semestre. Este chequeo constará de los siguientes puntos:

- Capacidad de almacenamiento de datos en la base de datos
- Limpieza de datos innecesarios almacenados en la base de datos
- Creación de un punto de restauración de datos
- Chequeo de posibles inconveniente reportados
- En caso de ser necesario, aplicar un recuperación de datos
- Cambios generales (nombres, etiquetas, tamaños)

También se dará mantenimiento vía e-mail, o por conversaciones en líneas telefónicas en caso de haber algún problema que requiera atención inmediata.

## CAPITULO 6: CONCLUSIONES Y RECOMENDACIONES

### 6.1 Conclusiones

#### 6.1.1 *Diseño*

- El diseño de un sistema estructurado por capas es complejo al inicio, ya que si no se tiene una idea bien clara de cuál es producto, se puede vagar en esta etapa por mucho tiempo.
- Realizar charlas periódicas con el cliente, acerca del avance del proyecto y recolección de nuevos requerimientos ayuda al desarrollo efectivo del mismo.
- La selección de las herramientas y técnicas adecuadas, según la magnitud o condición del proyecto, es de mucha ayuda para reducir tiempos de entrega y en caso de ser necesario, reducir costos de producción.

#### 6.1.2 *Desarrollo*

- El desarrollo de la aplicación se torna más fácil si tenemos un diseño robustamente estructurado. En ese caso, lo único necesario es seguir los pasos de desarrollo.
- La implementación de cada capa depende mucho de los diagramas UML realizados, de este modo, se logra entender la vinculación entre objetos y las dependencias necesarias en cada una.
- Es muy importante el observar el proceso que se sigue en la organización para determinar los requerimientos que se necesitan para la elaboración de un sistema y para ello se recurren a varias técnicas de recopilación de información para que el proyecto satisfaga las necesidades de los usuarios finales que es el que está en pleno contacto con él.

#### 6.1.3 *Soporte*

- La interacción constante con el cliente, facilita la etapa de soporte del sistema, ya que el cliente sabe lo que se hizo y que cosas específicas son las que requiere ayuda.
- El tiempo de resolución de problemas durante el desarrollo fue corto, ya que el cliente se encontraba físicamente cerca al desarrollador.

## 6.2 Recomendaciones

### 6.2.1 *Diseño*

- Es importante tener un esquema (maqueta) del sistema, que este bien estructurado, fácilmente entendible pero robusto; de esta manera, los riesgos al desarrollar se disminuirán.
- Es sumamente importante saber elegir las herramientas y metodologías adecuadas para el proyecto, antes de comenzar su desarrollo. Este proceso no hay que tomarlo a la ligera, ya que de él depende el éxito o fracaso del proyecto.

### 6.2.2 *Desarrollo*

- Se recomienda seguir los pasos de desarrollo tal como se estableció en el diseño, de tal forma que se cumplan los tiempos y capacidades de entrega. Entregar un producto incompleto a destiempo, reduce la confiabilidad prestada por el cliente.

### 6.2.3 *Soporte*

- Es bueno que exista un ambiente de confianza entre el desarrollador y el cliente, ya que de este modo, los problemas pueden ser tratados como temas de mejora y no como temas para atacar al producto desarrollado. La confianza es clave en todos los procesos de desarrollo del sistema.

## BIBLIOGRAFIA

### ***CAPITULO 1:***

En Internet:

PUCE, PLAN ESTRATEGICO DE DESARROLLO INSITUCIONAL 2008-2013.  
Internet. [http://www.puce.edu.ec/sitios/plan\\_estrategico](http://www.puce.edu.ec/sitios/plan_estrategico) (30/08/2010)

### ***CAPITULO 2:***

En Internet:

CALERO SOLIS, MANUEL. UNA EXPLICACIÓN DE LA PROGRAMACIÓN  
EXTREMA. Internet. <http://www.willydev.net/descargas/prev/explicaxp.pdf> (15/09/2010).

BEAS, JOSE MANUEL. PROGRAMACION EXTREMA. Internet.  
[http://es.wikipedia.org/wiki/Programaci%C3%B3n\\_extrema](http://es.wikipedia.org/wiki/Programaci%C3%B3n_extrema) (15/09/2010).

WELLS, DON. XP, A GENTLE INTRODUCTION. Internet.  
<http://www.extremeprogramming.org/> (15/09/2010).

ETON DIGITAL. RAPID APPLICATION DEVELOPMENT. Intenet.  
<http://www.etondigital.com/services/rapid-application-development/> (15/09/2010).

CASEMAKER INC. WHAT IS RAPID APPLICATION DEVELOPMENT?. Internet.  
[http://www.casemaker.com/download/products/totem/rad\\_wp.pdf](http://www.casemaker.com/download/products/totem/rad_wp.pdf) (15/09/2010).

TECHNOLOGY RESOURCES. JOINT APPLICATION DEVELOPMENT. Internet.  
[http://es.wikipedia.org/wiki/Joint\\_application\\_development](http://es.wikipedia.org/wiki/Joint_application_development) (15/09/2010).



MCCONELL, STEVE. JAD (RAPID APPLICATION DEVELOPMENT). Internet.  
<http://es.debugmodeon.com/articulo/jad-joint-application-development> (15/09/2010).

VIERKON ® 2003. (CADENA DE VALOR). Internet  
<http://www.vierkon.com/cadvalor.htm> (15/09/2010).

TODOEEXPERTOS.COM (MAPA DE PROCESOS). Internet.  
<http://www.todoexpertos.com/categorias/negocios/gestion-de-calidad/respuestas/937944/mapa-de-procesos> (15/09/2010).

ARIEL GOLDVARG, SIPOC - HERRAMIENTA PARA LA MEJORA DE PROCESOS. Internet.  
<http://www.discover6sigma.org/d6slab/sipoc/> (15/09/2010).

LAMARCA LAPUENTE, MARIA JESUS. BASES DE DATOS. Internet.  
[http://www.hipertexto.info/documentos/b\\_datos.htm](http://www.hipertexto.info/documentos/b_datos.htm) (15/09/2010).

PEREZ VALEZ, DAMIAN. QUE SON LAS BASES DE DATOS?. Internet.  
<http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-de-datos/>  
(15/09/2010).

CREATIVE COMMONS. LENGUAJES DE PROGRAMACION. Internet.  
<http://es.kioskea.net/contents/langages/langages.php3> (15/09/2010).

U. TECNOLOGICA, ARGENTINA. LENGUAJES DE PROGRAMACION. Internet.  
<http://frit.utn.edu.ar/sistemas/paradigmas/lenguajes.htm> (22/09/2010).

INSTITUTO NACIONAL DE ESTADISTICAS E INFORMATICA. HERRAMIENTAS CASE. Internet. <http://www.inei.gob.pe/biblioineipub/bancopub/Inf/Lib5103/Libro.pdf> (22/09/2010).

UNIVERSIDAD POLITECNICA DE VALENCIA. INTRODUCCION A HERRAMIENTAS CASE Y SYSTEM ARCHITECT. Internet. [http://users.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro\\_case\\_SA.pdf](http://users.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro_case_SA.pdf) (22/09/2010).

JOHN WILEY. INTRODUCTION TO CLIENT / SERVER SYSTEMS: A PRACTICAL GUIDE FOR SYSTEMS PROFESSIONALS. Internet. <http://temariotic.wikidot.com/la-arquitectura-cliente-servidor> (22/09/2010).

GALLI, RICARDO. DESARROLLO WEB EXTREMO. Internet. <http://bulma.net/body.phtml?nIdNoticia=734&nIdPage=2> (24/09/2010).

KERNEL ERROR. ARQUITECTURA 3 CAPAS. Internet. <http://kerneleerror.net/programacion/php/arquitectura-3-capas/> (24/09/2010).

## ***CAPITULO 5:***

En internet:

APACHE, SUBVERSION PRJECT. Internet. <http://subversion.apache.org/> (26/10/2010)

SELENIC, MERCURIAL PRJECT. Internet. <http://mercurial.selenic.com/> (26/10/2010)



## **GLOSARIO**

### **ASU**

Acción Social Universitaria.

### **Diagrama de Ishikawa**

Diagrama que permite determinar causas que disminuyen la eficiencia y rendimiento.

### **XP**

Extreme Programming, metodología de desarrollo de aplicaciones.

### **RAD**

Rapid Application Development, metodología de desarrollo de aplicaciones.

### **JAD**

Joint Application Development, metodología de desarrollo de aplicaciones.

### **Deployamiento**

Nombre ingles para implementación, que significa .ejecutar una aplicación empaquetada, lista para su uso.

### **CASE**

Computer-Aided Systems Engineering, Ingeniería en Sistemas ayudada por computadoras.

### **Flujograma**

Diagrama que muestra la secuencia de un proceso.

### **Java**

Lenguaje de programación interpretado por compilación.

### **Cliente**

En computación, equipo que pide servicios al servidor.

### **Servidor**





En computación, equipo que entrega servicios a los clientes.

### **Middleware**

En computación, equipo que procesa peticiones entre el servidor y el cliente.

### **Prototipo ejecutable**

Mini programa funcional usado como base para el desarrollo del producto final.

### **MySql**

Servidor de bases de datos gratis, conocida por su velocidad de respuesta.

### **Diagrama de Casos de Uso**

Diagrama que muestra los usos del sistema y los casos en los que se usa.

### **Clase**

Objeto que representa una entidad en la vida real.

### **Diagrama de Clases**

Diagrama que muestra la relación entre las clases del sistema.

### **JDK**

Java Development Kit, grupo de herramientas de desarrollo para JAVA.

### **Subversion**

Herramienta para control de versión de fuentes

### **Mercurial**

Herramienta para control de versión de fuentes